



IES Ciudad Escolar

**TRABAJO FINAL DE CICLO SUPERIOR DE DESARROLLO DE
APLICACIONES WEB**

SISTEMA DE GESTIÓN DE ESPACIOS COMUNES

Autores:

María García Valero
Gerardo Pimentel Serrano
Jossue Buenaño Peña

Tutora:

María Victoria González

Madrid, 03 de Junio del 2022

Índice

1. Resumen	2
2. Abstract	4
3. Justificación	5
4. Introducción	6
5. Objetivos	7
6. Temporalización, planificación y costes	8
7. Desarrollo del proyecto	10
7.1 Análisis y diseño	11
7.2 Implementación	23
7.3 Despliegue y mantenimiento	31
8. Conclusiones, mejoras futuras	33
9. Bibliografía y referencias	34
10. Anexos	35
Glosario de términos	35
Anexo 1 (Boceto):	36
Anexos 2 (Maqueta digital):	39
Anexo 3 (Versión 1 de la aplicación):	43
Anexo 4 (Versión 2 de la aplicación):	46
Anexo 5 (Versión final de la aplicación):	50
Anexo 6 (Cartel con Qr para reservar un aula):	55
Anexo 7 (Diagrama de Gantt):	56
Anexo 8 (Manual de usuario):	60
Anexo 9 (Manual de gestor):	60
Anexo 10 (Accesibilidad “Lighthouse”):	60

1. Resumen

En el presente trabajo fin de grado se realizará el diseño e implementación de una solución software con acceso vía web, cuya funcionalidad será implementar un sistema de gestión de espacios comunes para el IES Ciudad Escolar, el cual permitirá una sencilla gestión de datos maestros y las funcionalidades necesarias para la gestión de reservas.

Para ello se ha empleado el modelo de prototipado, el cual es un modelo de desarrollo de software en el cual el prototipo debe de ser construido rápidamente y con la utilización escasa de recursos. El prototipo se ha construido para mostrárselo al cliente, obtener críticas y retroalimentación, con lo cual se han obtenido los requisitos específicos para la aplicación a partir de las metas gráficas que han sido mostradas.

Para el desarrollo de la aplicación web se ha implementado un patrón de diseño MVC (Modelo-Vista-Controlador), creando así un sistema modular. Dentro de este, se ha empleado PHP como lenguaje base de programación, de igual manera haciendo uso de otras tecnologías como HTML5, CSS3, JavaScript y frameworks como Bootstrap 5.

Para poder almacenar los datos necesarios para llevar a cabo este proyecto, se ha empleado la herramienta XAMPP, ya que hemos utilizado Apache como servidor web, MySQL como sistema de base de datos y XAMPP nos provee las anteriores herramientas y de igual manera un intérprete de PHP, el cual es nuestro principal lenguaje de programación.

2. Abstract

In this final degree project we will design and implement a software solution with web access, whose functionality will be to implement a common spaces management system for the IES Ciudad Escolar, which will allow a simple management of master data and the necessary functionalities for the management of bookings.

For this purpose, the prototyping model has been used, which is a software development model in which the prototype must be built quickly and with little use of resources. The prototype has been built in order to show it to the customer, to obtain criticism and feedback, thus obtaining the specific requirements for the application from the graphical goals that have been shown.

For the development of the web application, an MVC (Model-View-Controller) design pattern has been implemented, thus creating a modular system. Within this, PHP has been used as the basic programming language, as well as making use of other technologies such as HTML5, CSS3, JavaScript and frameworks such as Bootstrap 5.

In order to store the necessary data to carry out this project, the XAMPP tool has been used, as we have used Apache as a web server, MySQL as a database system and XAMPP provides us with both the aforementioned tools and also a PHP interpreter, which is our main programming language.

3. Justificación

La realización de este proyecto viene dada por la necesidad de la dirección del centro IES Ciudad Escolar de llevar a cabo una organización para las reservas de los espacios comunes del centro y así poder realizar estas reservas de una forma sencilla y ordenada. Con la introducción de esta aplicación el centro obtendrá un gestor de reservas así como una interfaz sencilla para que los profesores puedan realizar las reservas.

La aplicación se va a desarrollar en gran parte con el lenguaje servidor PHP aunque también se realizarán partes con el lenguaje JavaScript, HTML y CSS. Hemos decidido usar el lenguaje de PHP puesto que es uno de los más utilizados hoy en día debido a su utilidad y versatilidad.

4. Introducción

Este documento describe el trabajo realizado por María García, Jossue Buenaño y Gerardo Pimentel en el proyecto final del ciclo formativo de grado superior de Desarrollo de aplicaciones Web, está dirigido a los profesores de este ciclo.

Este proyecto consiste en la realización de un sistema de gestión de espacios comunes elaborado para el IES Ciudad Escolar. La aplicación consta de varias interfaces principales, cuyo aspecto y funcionalidad depende del rol del usuario registrado.

La aplicación se desarrolla en web, de forma que los usuarios podrán acceder a ella sin necesidad de instalar software en su equipo informático y con el único requisito de tener un navegador web y una conexión a internet, lo cual a día del desarrollo de este trabajo fin de grado está muy extendido en nuestro país y el mundo.

5. Objetivos

En el siguiente trabajo se pretende la realización de un sistema de gestión de espacios comunes cuyos objetivos son:

- Facilitar a los usuarios, en este caso, los profesores del IES Ciudad Escolar, la realización de reservas de las aulas comunes.
- Facilitar al gestor, en este caso, el jefe de estudios y/o dirección, la cancelación y modificación de reservas así como la creación, modificación y eliminación de aulas.
- Realizar un análisis anual de las aulas, de forma que estas estadísticas sirvan al centro para la mejora del mismo.
- Creación de roles para que el propio gestor pueda crear nuevos roles permitiendo que los usuarios, en función de su rol, realicen las funciones que el gestor requiera.
- Parametrización de la aplicación de forma que se puedan añadir festivos, se puedan bloquear aulas específicas, se pueda modificar la cantidad de días de margen con el que se puede reservar... se trata de realizar una aplicación que no sea estática para que el gestor la pueda adaptar a sus necesidades del momento.

6. Temporalización, planificación y costes

En cuanto a los **costes** el precio medio de un programador freelance en España es de 20-45€/h, por lo que hemos decidido cobrar un precio medio de 45€/h por cada miembro del equipo.

Teniendo en cuenta que somos 3 programadores y el tiempo estimado de desarrollo es de 30 horas, el coste del proyecto es de 1.350€.

En cuanto a la **temporalización y planificación** hemos dividido el proyecto en fases, teniendo en cuenta el ciclo de vida del software:

- La primera fase es la de análisis y diseño, al ser una fase en la que solo se piensa la estructura de la aplicación y se hacen diagramas orientativos consideramos que nos va a llevar un tiempo de una semana.
- La segunda fase es la de desarrollo, es la fase más densa puesto que es cuando realizamos toda la aplicación, en esta fase habrá partes que se desarrollen de forma conjunta y otras en las que nos dividiremos, iremos realizando reuniones cada dos-tres días para poder valorar cómo va la aplicación e ir dividiendo las tareas o rehaciendo aquello que esté mal, al ser la fase más importante y complicada hemos estimado que nos llevará unas siete semanas.
- La tercera fase es la de pruebas, en esta fase realizaremos pruebas de seguridad y usabilidad, al tratarse de probar la aplicación vamos a dividirnos para realizar distintas pruebas de forma más detallada, estimamos una semana de trabajo para llevar a cabo esta fase.
- La cuarta fase es la de despliegue, estimamos que nos va a llevar una semana de tiempo y la realizaremos entre todos.

Importante recordar que utilizaremos un **ciclo de vida ágil**, por lo que estas fases se podrán repetir a lo largo del desarrollo del proyecto.

La documentación del presente documento se realizará a lo largo de todo el tiempo, por lo que no se engloba en ninguna fase.

En el siguiente Diagrama de Gantt hemos realizado una aproximación sobre los tiempos y actividades que vamos a realizar cada uno, aunque es orientativo puesto que el tiempo real que nos va a llevar cada tarea o las tareas que puedan surgir no lo sabemos.

Hemos dividido nuestro tiempo del proyecto en 4 fases, a continuación mostramos el detalle de la primera fase, si se quiere ver al completo el Diagrama de Gantt con todas las fases se encuentra en el anexo 7. (También se adjunta el documento excel para poder visualizarlo mejor).

ACTIVIDAD	Sistema de Gestión de Espacios Comunes								
	1	2	3	4	5	6	7	8	9
FASE 1									SEMANAS de 14/03/22 a 03/06/22
Análisis y diseño									
Análisis de requisitos	Maria								
División de requisitos en tareas	Gerardo								
Diseño inicial de la aplicación	Jossue								
Desarrollo e implementación									
Desarrollo de estructura del código		Gerardo							
Desarrollo de base de datos		Jossue							
Desarrollo de vistas		Maria							
Pruebas									
Pruebas de seguridad y funcionamiento de la fase 1			Gerardo y María						
Pruebas de accesibilidad y usabilidad			Jossue						
Documentación		María, Gerardo y Jossue							
Desarrollo 30% de la documentación									
Entrega prototipo fase 1				Entrega F1					

7. Desarrollo del proyecto

El proyecto se puede encontrar [en nuestro repositorio de GIT](#):

<https://github.com/GerardoJossueMaria/SGEC>

Se incluye en los anexos un glosario de términos para definir aquellos conceptos más técnicos.

Para el desarrollo de la aplicación hemos utilizado Visual Studio Code como principal entorno de desarrollo.

Para el desarrollo de la aplicación hemos utilizado:

- **HTML:** Estructura de la aplicación.
- **MySQL:** Base de datos.
- **CSS:** Estilos de la aplicación.
- **PHP:** Conexión entre servidor y cliente.
- **JavaScript:** Añadir características interactivas a la aplicación.
- **Xampp:** Software libre, sistema de gestión de bases de datos MySQL, el servidor web Apache y los intérpretes para lenguajes de script PHP.
- **AJAX:** Comunicación dinámica entre servidor y resultados de la barra de búsqueda sin necesidad de actualizar la página para ofrecer una mejor experiencia al usuario.
- **Jquery:** Librería de JS que permite facilitar el uso de JS.
- **SWAL Alert:** Plugin de Jquery para personalizar mensajes de alerta.
- **Bootstrap:** Facilitar partes del diseño de la interfaz de la aplicación.
- **Bootstrap icons:** Biblioteca de iconos gratuita.
- **Canva:** Herramienta para diseño de interfaz.
- **Git:** Control de versiones del proyecto y repositorio para poder trabajar todos conjuntamente.
- **CDMON:** Plataforma para realizar el despliegue de la aplicación.
- **Lighthouse:** Extensión de Google Chrome para comprobar la accesibilidad.

7.1 Análisis y diseño

Para comenzar el proyecto lo primero que necesitamos es la **expresión de necesidades** por parte del cliente, estas necesidades y requisitos del cliente se han formalizado en un documento (se podrá encontrar en el repositorio de git) para así nosotros poder continuar con la siguiente fase del desarrollo de nuestro proyecto.

Una vez tenemos el documento de las necesidades del cliente avanzamos a la fase 2 que es la **planificación y análisis** de los requisitos del cliente para poder comprobar cuales están poco claros, incompletos...y así aclarar las dudas antes de comenzar a desarrollar nuestro proyecto.

Dentro del análisis de nuestro proyecto, una de las partes más importantes es estudiar la viabilidad del proyecto, puesto que podrían pedirnos funcionalidades que se encuentran fuera de nuestro alcance, y así poder planificar el tiempo de duración del proyecto e identificar las limitaciones para ello vamos a extraer aquellos requerimientos para poder analizarlos de forma más detallada, los requisitos se dividen en:

- **Requisitos funcionales:** describen cómo debe comportarse la aplicación.
- **Requisitos no funcionales:** especifica criterios que pueden usarse para juzgar la operación de un sistema en lugar de sus comportamientos específicos.

Para el desarrollo de nuestra aplicación hemos extraído los siguientes **requisitos**:

- Consulta diaria de todos los espacios disponibles (Crítico).
- Consulta semanal por espacio (Crítico).
- Reserva de espacio desde cualquier vista (Crítico).
- Anulación de una reserva de espacio (Crítico).
- Reserva prioritaria de espacio por parte de un gestor (Crítico).
- Alta de un nuevo espacio (Crítico).
- Modificación de un espacio (Crítico).
- Baja de un nuevo espacio (Crítico).
- Modificación del tiempo máximo de antelación de la reserva (Crítico).
- Alta de un PEC (Crítico).
- Modificación de un PEC (Crítico).

- Baja de un PEC (Crítico).
- Interfaz adaptada al rol del usuario (Crítico).
- Uso de colores asociados al estado de la reserva (Crítico).
- Interacción con la GUI a través de ratón y teclado (Crítico).
- Metainformación asociada a imágenes (Crítico).
- Máxima concurrencia de acceso (Crítico).
- Tiempos de respuesta (Crítico).
- Implementación de roles en la aplicación (Crítico).
- Longitud y complejidad de las claves (Crítico).
- Cifrado de las claves (Crítico).
- Implementación HTML (Crítico).
- Implementación CSS (Crítico).
- Implementación PHP (Crítico).
- Implementación SQL (Crítico).
- Gestión de calendario (festivos) (En previsión).
- Estadísticas de ocupación (En previsión).
- Interfaz web adaptativa o “responsive” (En previsión).
- Listado de reservas por día (Opcional).
- Baja sin borrado de un nuevo espacio (Opcional).
- Baja sin borrado de un usuario (Opcional).
- Formato compacto de visualización/impresión (Opcional).
- Interacción con el sistema de reservas a través de un ratón exclusivamente (Opcional).
- Interacción con el sistema de reservas a través de un teclado exclusivamente (Opcional).
- Bloqueo tras 3 intentos consecutivos de acceso con clave incorrecta (Opcional).

Estos requerimientos han sido resumidos en:

Requisito	Funcional/No funcional	Descripción
RE1	Funcional	Reserva de aulas
RE2	Funcional	Vista mensual
RE3	Funcional	Vista semanal
RE4	Funcional	Vista diaria

RE5	Funcional	Gestión de usuarios
RE6	Funcional	Gestión de aulas
RE7	Funcional	Gestión de roles
RE8	Funcional	Gestión de reservas
RE9	Funcional	Gestión de grupos
RE10	Funcional	Gestión de reservas (cada usuario las suyas)
RE11	Funcional	Actualización de base de datos
RE12	Funcional	Carga de nuevos contactos
RE13	Funcional	Estadísticas
RE14	No Funcional	Intuitiva
RE15	No Funcional	Uso fácil
RE16	No Funcional	Fácil mantenimiento
RE17	No Funcional	Envío/respuesta de datos rápido
RE18	Funcional	Modificación de nº días a reservar
RE19	Funcional	Cambio por parte del usuario de sus datos
RE20	No funcional	Bloqueo tras 3 intentos de iniciar sesión
RE21	No funcional	Complejidad de la clave

Tabla resumen de los requisitos del proyecto

Una vez establecidos los requisitos necesitamos especificar qué tareas vamos a realizar para que sea más sencilla la organización del equipo:

Tarea	Descripción	Requisito
TA1	Calendario mensual	RE1, RE2, RE4
TA2	Vista semanal	RE1,RE3
TA3	Gestionar Usuarios, Aulas, Roles, Grupos y Reservas	RE5,RE6,RE7,RE8,RE9
TA4	Gestión de reservas de cada usuario	RE10
TA5	Actualización de base de datos	RE11
TA6	Estadísticas	RE13
TA7	Carga de nuevos usuarios	RE12
TA8	Usabilidad y accesibilidad	RE14,RE15,RE16,RE17
TA9	Configuraciones	RE18, RE19
TA10	Pruebas	
TA11	Despliegue	

Con todos los requisitos analizados y claros, las tareas definidas, y las limitaciones establecidas, comenzamos la fase de **diseño**, en esta fase ya comenzamos a visualizar la solución con la ayuda de las anteriores fases. Para poder obtener una visión más completa de toda la aplicación realizamos diversos diagramas para poder obtener un mayor detalle de la funcionalidad e interfaz que vamos a desarrollar.

El primer diagrama que hemos realizado es el de casos de uso, para poder analizar la parte funcional, con este diagrama tratamos de entender qué funciones debe realizar nuestra aplicación.

Diagrama de casos de uso

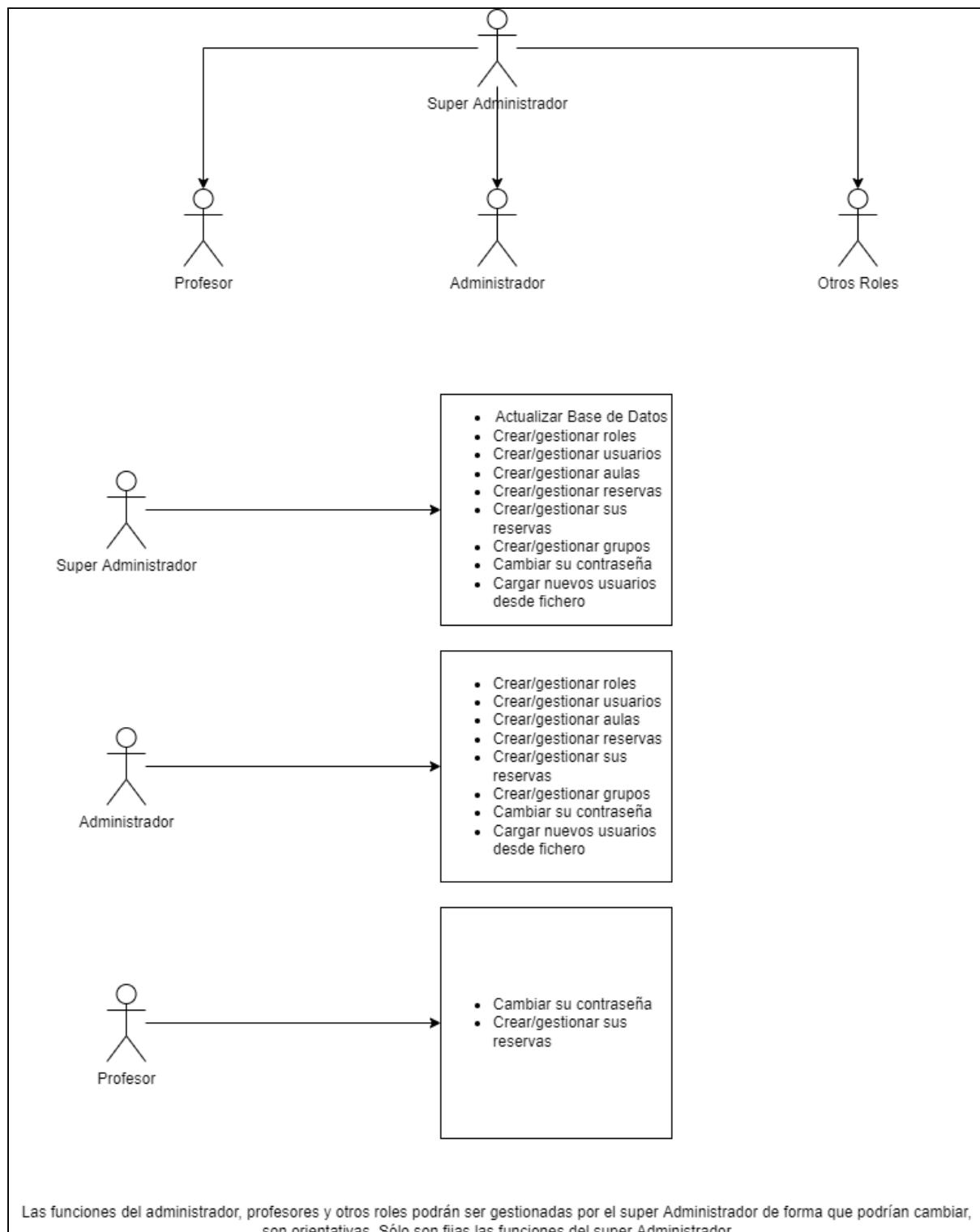


Diagrama de casos de usos del SGEC

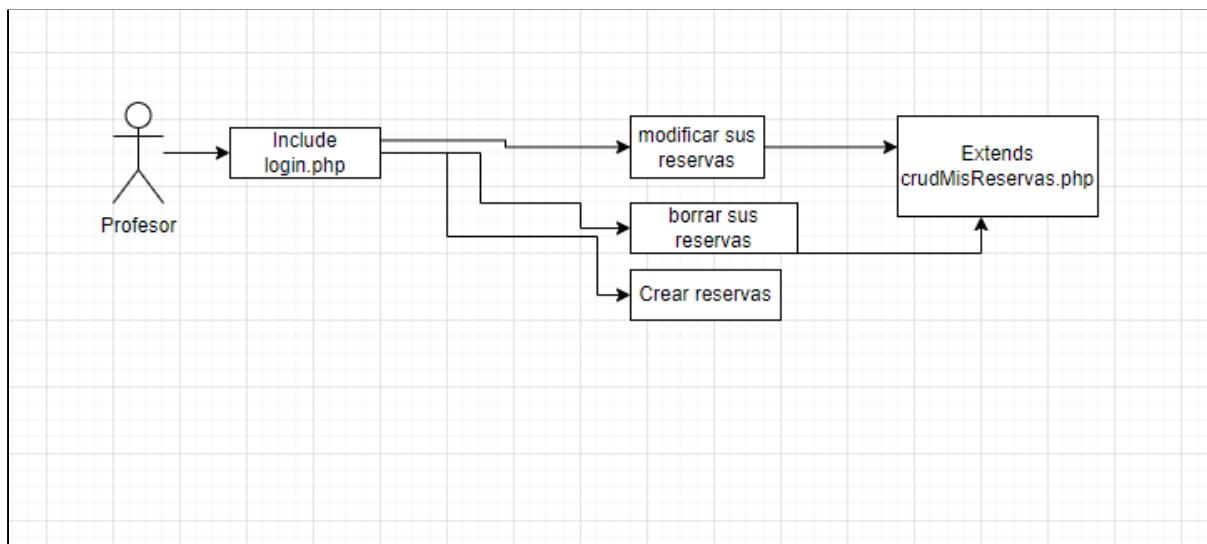


Diagrama de casos de usos de los profesores del SGEC

El siguiente diagrama que hemos realizado es un diagrama de flujo para poder analizar cómo vamos a desarrollar la parte de reservas.

En los requisitos del cliente se nos pidió que los usuarios puedan reservar con dos interfaces distintas (vista mensual y vista semanal) por lo que hemos hecho un diagrama para el desarrollo de cada interfaz:

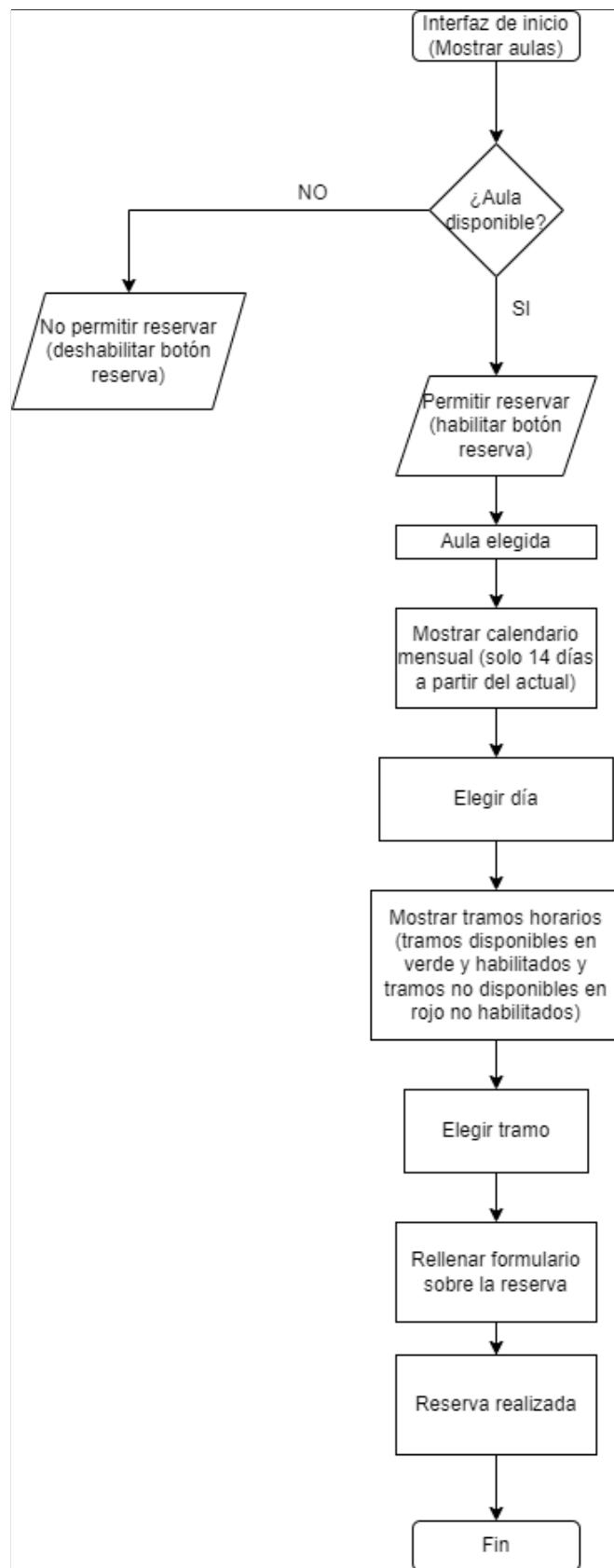
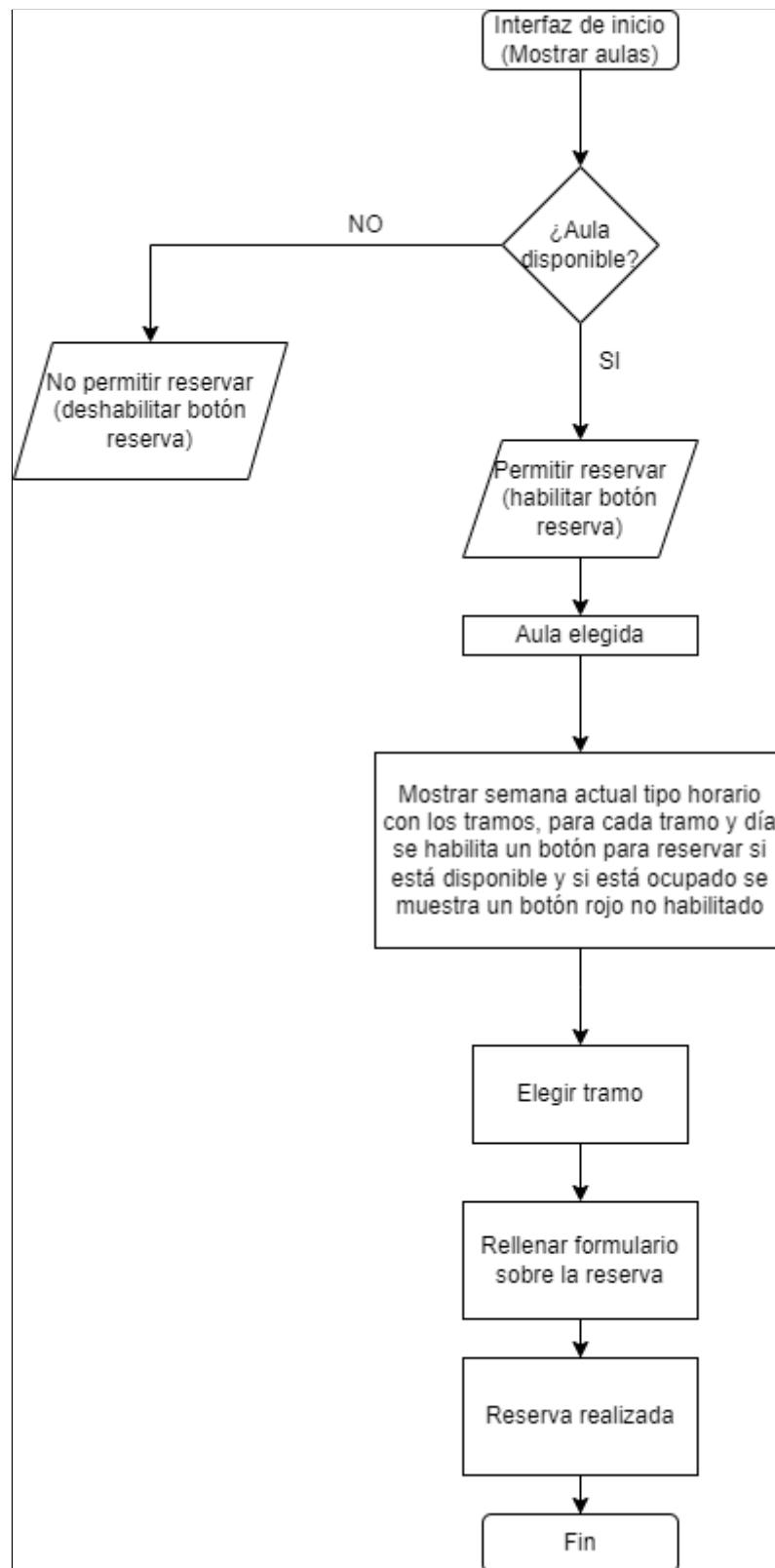
Diagrama de flujo para la realización de una reserva en el calendario mensual:

Diagrama de flujo para la realización de una reserva en el calendario semanal:

Con todas las funcionalidades definidas, decidimos que para esta aplicación el mejor patrón de diseño es el **Modelo-Vista-Controlador** que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos.

Se trata de un modelo que ha demostrado su validez a lo largo de los años en todo tipo de aplicaciones, y sobre multitud de lenguajes y plataformas de desarrollo.

- El Modelo contiene una representación de los datos que maneja el sistema, su lógica de negocio, y sus mecanismos de persistencia.
- La Vista compone la información que se envía al cliente y los mecanismos de interacción con éste.
- El Controlador, que actúa como intermediario entre el Modelo y la Vista, gestiona el flujo de información entre ellos y las transformaciones para adaptar los datos a las necesidades de cada uno.

Para poder tener una idea más visual de la arquitectura de nuestra aplicación hemos creado un diagrama de clases:

Diagrama de clases

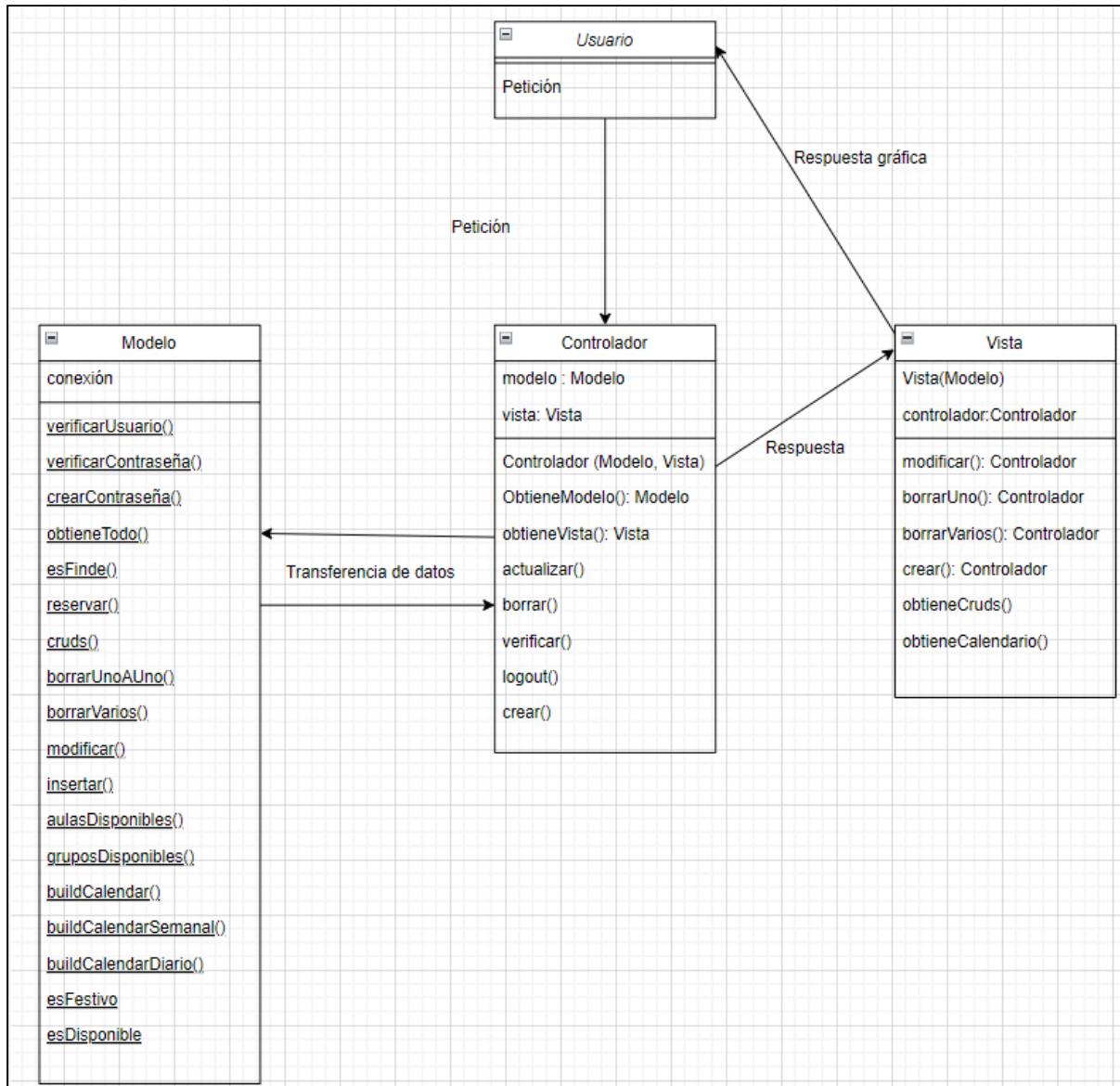
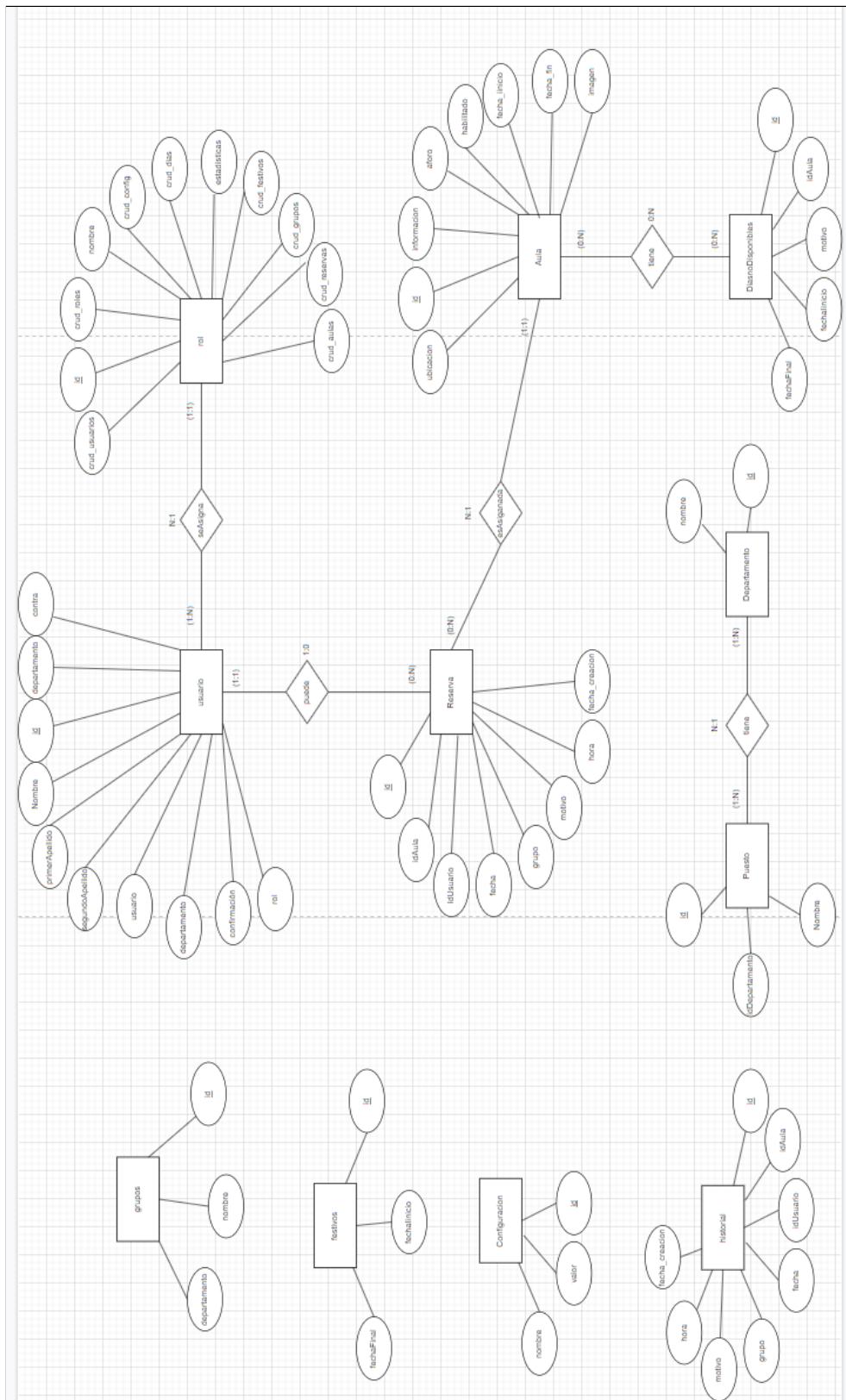


Diagrama de clases de la estructura del código

Una vez tenemos la parte funcional definida, pasamos la creación de un primer diseño (Anexo 1 y 2) de la interfaz, se trata de un diseño orientativo para posteriormente desarrollar y mostrar al cliente para realizar los cambios que estime oportunos.

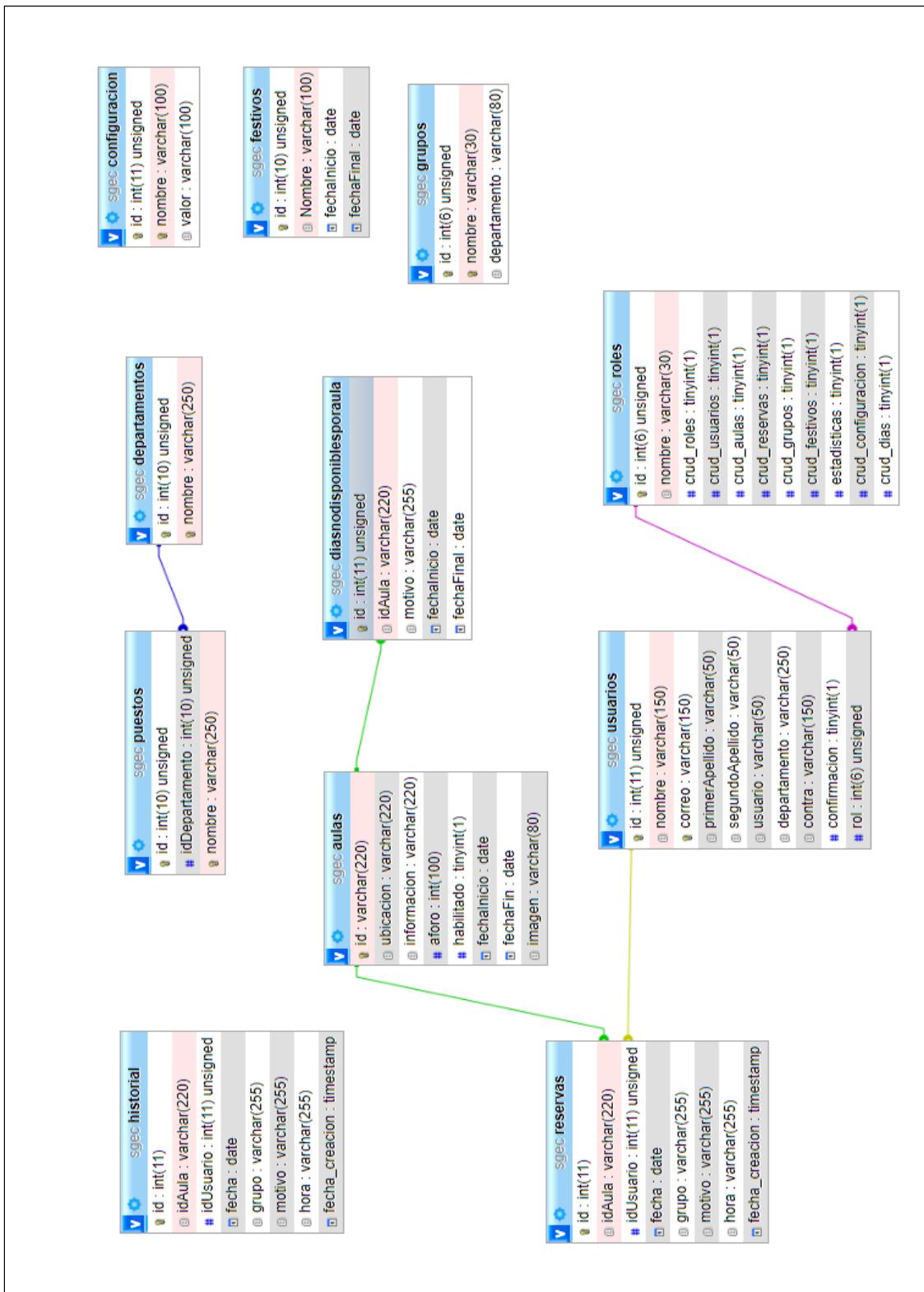
Por último para poder finalizar la fase del diseño y dar comienzo al desarrollo de la aplicación creamos un diseño de la base de datos que vamos a utilizar, primero realizamos un modelo entidad-relación para después pasar a un modelo relacional:

Modelo Entidad-Relación



Modelo entidad-relación de la BBDD

Modelo Relacional



Modelo relacional de la BBDD

7.2 Implementación

Finalizada la parte del diseño damos comienzo a la fase de **implementación**, en esta fase elegimos el lenguaje de programación del que poseemos más conocimientos, en nuestro caso hemos elegido realizar la aplicación con PHP como lenguaje principal aunque también usaremos JavaScript para algunas partes.

Una vez desarrollada la aplicación pasamos a mostrársela al cliente para poder ir corrigiendo los errores y realizando los cambios necesarios, esto se conoce como metodología Agile, que es la metodología que hemos decidido utilizar ya que es la que más se adapta a la forma de realizar nuestra aplicación y que consiste en:

- Las personas y las interacciones son la prioridad siempre, incluso por encima de los procesos y las herramientas.
- Que el producto funcione perfectamente es más importante que una documentación extensa al respecto.
- Colaborar con el cliente siempre será preferible a establecer una negociación contractual.
- El proceso de trabajo siempre debe responder ante el cambio, y nunca enfocarse en un plan estricto.

En este caso el cliente nos pidió que la carga de nuevos usuarios no elimine ni vuelva a insertar los que ya están en la base de datos, nos pidió realizar una parte de gestión de festivos para evitar la reserva los días festivos, y definió las estadísticas sobre el uso de aulas que vamos a crear.

Una vez resueltos estos requisitos obtenemos un PMV (Producto Mínimo Viable). Lo primero que hemos realizado es la vista de las reservas ya que es la parte más importante de nuestra aplicación, para ello hemos creado 3 vistas distintas aunque todas funcionan de forma muy similar, con consultas a la base de datos.

También hemos creado diferentes tablas para que el administrador pueda modificar los datos de la base de datos de una forma más sencilla e intuitiva.

También hemos querido añadir que el usuario pueda modificar su contraseña o una parte de ayuda con varias preguntas que les puede surgir a nuestros usuarios.

Para el desarrollo de la aplicación cabe destacar algunas de las funciones que el cliente no ha pedido pero hemos decidido añadir como por ejemplo: hemos creado una función que al reservar envía un correo de confirmación al usuario con los datos de la reserva, esto se ha realizado con la función PHPMailer, una barra de búsqueda para que tanto al gestor como a los usuarios profesores les sea más fácil localizar una reserva concreta, un usuario concreto etc y también hemos realizado que el parámetro de los días máximos con los que se puede reservar sea modificable.

Para obtener las estadísticas hemos creado una tabla “historial” que, con ayuda de un evento en MySQL, cada día elimina las reservas del día anterior y las almacena en esta tabla de forma que el crud de reservas se muestre más “limpio” solo con las reservas actuales y las estadísticas se realizan sobre esta tabla historial”.

Uno de los requisitos es que si el usuario intentaba acceder más de 3 veces de forma incorrecta se bloqueara, para ello lo que hemos hecho finalmente es que si el usuario intenta acceder más de 3 veces de forma incorrecta se bloquea su cuenta durante 30 segundos, el control de los intentos lo hemos realizado guardando una cookie en el cliente para evitar ataques de bots y así darle mayor seguridad.

Para comprobar la accesibilidad de nuestra web hemos utilizado la extensión de google chrome “Lighthouse” que nos indica el porcentaje de accesibilidad y aquellos aspectos que se pueden mejorar y en nuestro caso nos indica que nuestra aplicación es 100% accesible (Anexo 10).

Y por último destacamos la parte de “Recordar contraseña”, esta parte la hemos realizado de forma que el usuario introduce su correo electrónico, si el correo no existe se muestra un mensaje de error, si el correo existe se envía una contraseña temporal creada aleatoriamente y se le pide que inicie sesión con esa contraseña, una vez inicia sesión se le pide que introduzca una nueva contraseña.

En cuanto al uso de la base de datos, no sólo almacenamos la información, también hemos creado funciones para validar la contraseña y el usuario, un procedimiento para el cambio de contraseña y un evento para la realización de las estadísticas.

Cabe destacar las siguientes funcionalidades:

- Con la tecnología AJAX hemos utilizado para la creación de las barras de búsqueda, para poder realizar llamadas al servidor de forma dinámica y sin que el usuario tenga que recargar la página, ofreciendo un diseño mucho más limpio y profesional.
- Con la librería SWAL de JQuery la hemos implementado a la hora de crear todos los mensajes de alerta, ya que permite que los mensajes sean visualmente más llamativos.
- Hemos creado un CRONJOB (evento), como hemos mencionado anteriormente, para guardar el historial de las reservas de una forma más profesional y sencilla.
- Todos los formularios de acceso a la aplicación están comprobados tanto con HTML, PHP y JS, de forma que la aplicación es mucho más segura.
- Se puede poner como “no disponible” un aula en unas determinadas fechas.
- Se pueden bloquear la reserva de ciertos días y bloquear la reserva de los días festivos.
- Se puede cargar un fichero csv o txt para importar los nuevos usuarios insertando solo aquellos que no se encuentren ya en la base de datos.
- El gestor puede modificar el número de días máximos a reservar.
- Cada usuario, dentro de su perfil, puede cambiar su contraseña.
- Los usuarios van a encontrar un espacio de “Ayuda” en el que resolvemos las posibles dudas que puedan tener.
- La reserva de un aula puede realizarse desde 3 vistas diferentes (calendario mensual, calendario semanal y vista de todas las aulas por días).
- Puesto que nos hemos encontrado con el problema de que el gestor no puede importar todos los correos de los profesores hemos realizado una vista en la que si el usuario (profesor en este caso) no tiene correo y es la primera vez que inicia sesión, se le pedirá que ingrese su correo electrónico y la nueva contraseña.
- Si el usuario olvida su contraseña puede recuperarla en “He olvidado mi contraseña” de forma que se le enviará a su correo una contraseña aleatoria la cual se obliga a cambiar al iniciar sesión con ella.
- El usuario puede mantener su sesión iniciada si al iniciar sesión marca la opción de “Recordar contraseña”.
- Por seguridad la contraseña no se muestra mientras se escribe pero si el usuario pulsa en el ojo que aparece a la derecha del input, se le muestra.
- Si un mismo usuario intenta acceder a la aplicación 3 veces seguidas con datos incorrectos, se le bloquea el acceso durante 30 segundos, esta funcionalidad la hemos controlado mediante las cookies del usuario.

- Si al crear un aula el gestor no dispone de imagen para ese aula, se inserta una imagen de un aula por defecto.
- Puesto que borrar es una acción irreversible e importante, hemos realizado una confirmación de forma que si el usuario decide borrar en lote o borrar de forma individual un dato se le pedirá que confirme esta acción.
- Destaca la seguridad que hemos creado en casos como, por ejemplo, que al crear una reserva si la quieres modificar, esta reserva no te va a dejar cambiarla a un día festivo o fin de semana, todos los datos que se quieran modificar pero sus valores tengan que estar reflejados en otras tablas son desplegables, para que al gestor/usuario le sea más sencillo modificar los datos, todas las páginas están cifradas de modo que para acceder a ellas es necesario iniciar sesión y tener acceso, no se puede acceder a una página por URL si no tienes permiso.
- La aplicación es responsive, es decir, se puede acceder a ella desde cualquier dispositivo.
- Hemos creado unos carteles con un código QR por cada aula, de forma que los usuarios, si tienen la sesión iniciada, escaneando este código pueden acceder directamente a este aula (Anexo 6).
- Con la plataforma CDMON, hemos realizado un despliegue sencillo y seguro.
- Comprobación de accesibilidad con extensión “Lighthouse”.

Con la aplicación desarrollada pasamos a realizar las **pruebas** de seguridad correspondientes para evitar ataques y la introducción/modificación de datos de forma incorrecta. Para evitar que los usuarios introduzcan datos de forma incorrecta (por ejemplo, modificar una reserva y añadir una fecha superior a 14 días o anterior al día actual o en fin de semana) hemos controlado, antes de insertar en la base de datos, que los datos sean en el formato deseado, esto lo controlamos mediante estructuras de control IF con PHP, en caso de que el dato sea correcto se procederá a la inserción/modificación en la base de datos, si el dato no es correcto, con JS mostramos un mensaje al usuario para ayudarle a corregir los datos que están mal.

Para evitar ataques (por ejemplo que un usuario profesor acceda a una función de administrador), el acceso a todas las páginas de nuestra aplicación está restringido de forma que solo pueden acceder si el usuario tiene el rol que da acceso a esa parte de la aplicación.

Para evitar que un bots intente saturar el servidor a la hora de iniciar sesión el número de intentos de inicio de sesión se ha restringido con la base de datos mediante el navegador del usuario.

Todas las pruebas de seguridad que hemos realizado se engloban en 5 tipos:

- Revisión de código
- Pruebas unitarias
- Pruebas de integración
- Pruebas de sistemas
- Pruebas de usabilidad y accesibilidad

Con las primeras pruebas queremos evaluar el contenido de cada módulo y con las segundas pruebas tratamos el módulo como una caja y lo probamos con valores de entrada evaluando los valores de salida. Con las pruebas de usabilidad y accesibilidad comprobamos que el sistema sea accesible para todos los usuarios.

Revisiones de código

El objetivo de estas pruebas es revisar el código que hemos escrito para comprobar que es eficiente, para ello nos reunimos todos los miembros del equipo y vamos analizando las partes de la aplicación para valorar si es lo suficientemente eficiente.

Pruebas unitarias

Estas pruebas se realizan para controlar el funcionamiento de pequeñas partes de la aplicación, estas pruebas se realizan usando cada parte de la aplicación para poder comprobar si da error, por ejemplo, si comprobamos la parte de usuarios tenemos que verificar que se agregan bien los usuarios, que todos los campos al crear o modificar un usuario están llenos, que la parte de elegir el rol de un usuario es de tipo select para que el administrador no introduzca un rol inexistente...los tipos de pruebas que se realizan dependen de la estructura de cada parte, quedando de la siguiente forma:

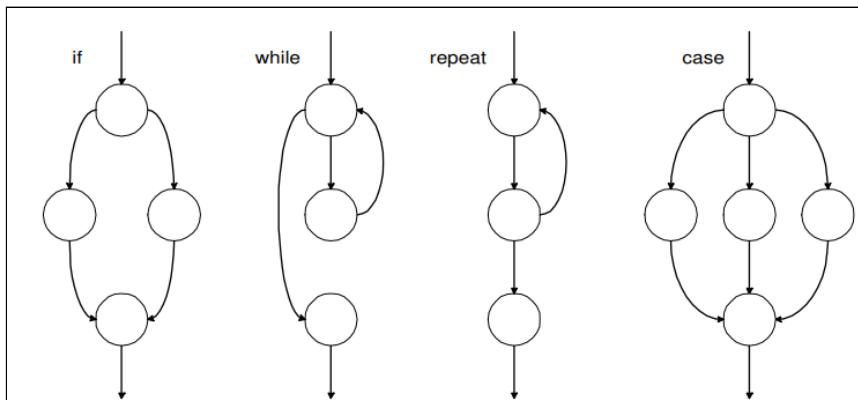


Diagrama de cómo se realizan las pruebas en función de la estructura de la función

Pruebas de integración

Al ya haber realizado las pruebas unitarias, estas pruebas se realizan como caja negra; Estas pruebas consisten en ir uniendo diferentes módulos e ir comprobando que funcionan una vez integrados hasta conseguir el conjunto de la aplicación. Existen dos tipos de integración: la integración incremental (se van integrando los módulos poco a poco) y la integración no incremental (se integran todos los módulos a la vez).

En nuestro caso la integración ha sido incremental ya que cada vez que realizamos un nuevo módulo lo integramos al sistema.

Cada vez que se integra un nuevo módulo se comprueba el funcionamiento de todo el sistema con los módulos actuales, para comprobar que la unión de este nuevo módulo no ha afectado al funcionamiento de los módulos ya integrados.

Pruebas de sistema

Estas pruebas se realizan una vez integrados todos los componentes, el objetivo de estas pruebas es comprobar la respuesta del sistema en conjunto frente a diversas situaciones, para ello simulamos varios escenarios y comprobamos su comportamiento. Para esta parte hemos realizado diferentes pruebas:

- Pruebas negativas: con este tipo de pruebas buscamos que nuestra aplicación falle para así poder comprobar y reforzar sus debilidades, para ello tratamos de introducir datos erróneos o realizar acciones que no se deberían poder realizar, por ejemplo, una vez iniciado sesión como profesor tratar de acceder a la tabla (crud) de los datos de usuarios.

- Pruebas de rendimiento: tratamos de comprobar el rendimiento de la aplicación, con este tipo de pruebas queremos comprobar la rapidez con la que el administrador puede modificar los datos o el tiempo que se tarda en realizar una reserva. Si el tiempo lo permite, se realizará una prueba de estrés dando acceso a todos los profesores del centro para ver como se comporta la aplicación con un gran número de usuarios entrando a la vez.
- Pruebas de seguridad: este tipo de pruebas es de las más importantes ya que tratamos de comprobar que nuestra aplicación es segura, para ello tratamos de acceder de diversas formas al sistema para comprobar que no hay errores, por ejemplo, no puedo acceder a ninguna página si no he iniciado sesión, para evitar ataques de bots controlamos el número de intentos de inicio de sesión mediante la ip y la base de datos...la parte que más hemos dotado de seguridad es el inicio de sesión ya que si un usuario no autorizado consiguiera entrar al sistema supondría un grave problema.

Pruebas de usabilidad y accesibilidad

Hasta este momento todas las pruebas que hemos realizado son funcionales, pero en las pruebas de usabilidad y accesibilidad nos centramos en el diseño de la aplicación, con este tipo de pruebas tratamos de hacer accesible nuestro sistema.

Para ello comprobamos que todas las imágenes llevan asociada meta información, comprobamos que el tamaño de la letra, botones...es el adecuado, los colores son distinguibles y se basan en lo “tradicional” (rojo=eliminar, verde=aceptar...), se puede acceder desde cualquier navegador, se puede acceder desde un dispositivo móvil...

Resumen de las pruebas realizadas:

Tipo de prueba	Prueba realizada
Revisión de código	Cada vez que implementamos una nueva función comprobamos que el código es eficiente poniendo en común nuestras opiniones acerca de cómo se ha realizado.
Pruebas unitarias	Todas las funciones del crud funcionan, se puede crear una reserva, se puede cambiar de vista, se puede cerrar sesión...

Pruebas de integración	Se puede navegar entre las pestañas, se puede cambiar de vista y en todas se muestran los datos de reservas de forma correcta, todos los crud's funcionan una vez unidos en la aplicación...
Pruebas de sistema	Tratamos de acceder a través de la URL a páginas en las que no tenemos permiso, realizamos una reserva con los mismos datos al mismo tiempo para comprobar que no falla, prueba de estrés para comprobar el rendimiento de la aplicación...
Pruebas de usabilidad y accesibilidad	La aplicación es interpretable por lectores de pantalla, se facilita la navegación a través del menú, el diseño es sencillo y limpio, el diseño es responsive, no se abusa de animaciones o colores, uso de iconos, alternativas textuales para contenido visual, mecanismos de contexto y orientación...

Con todas las pruebas realizadas, hemos comprobado que nuestra aplicación es lo suficientemente segura y accesible para todos los usuarios por lo que ya podemos pasar a la última fase.

7.3 Despliegue y mantenimiento

Para finalizar el desarrollo de nuestra aplicación, realizamos la última fase del proyecto la cual es el **despliegue**; para desplegar nuestro proyecto hemos utilizado el proveedor **CDMON**, que nos permite un despliegue de forma sencilla y gratuita durante un periodo de tiempo y, aunado a ello, es el proveedor de hosting predilecto de nuestro cliente.

Utilizando **CDMON** como solución **PaaS** (Platform-as-a-Service), hemos conseguido realizar esta fase a través de su intuitivo panel de control, el cual nos permite desplegar subiendo los ficheros de nuestro software local a carpeta “web” mediante de un servidor FTP Online.

A partir del momento en el que se encuentran los ficheros dentro del servidor FTP Online y asegurarnos de que funcionan correctamente, debemos migrar nuestra base de datos local a la nueva base de datos online, lo cual hemos realizado a través del panel de control que nos ofrece **phpMyAdmin** en el apartado “Exportar” para obtener un archivo con nuestra base de datos local.

Contando con el archivo de nuestra base de datos local, hemos utilizado el panel de control de CDMON para crear una base de datos online, la cual podemos gestionar mediante **phpMyAdmin** que nos ofrece CDMON. Una vez dentro (**con el usuario y la clave proveídas por CDMON**) debemos seleccionar la base de datos que hemos creado anteriormente y dirigirnos al apartado “Importar” en el cual debemos seleccionar el archivo con nuestra base de datos local.

Hechas las acciones anteriores, hemos conseguido desplegar satisfactoriamente tanto nuestro software como nuestra base de datos; pero todo esto se ha realizado sobre un hosting gratuito de pruebas, por lo que debemos ahora migrar toda nuestra web y base de datos a un nuevo usuario que posea un contrato de hosting pago para poder registrar el dominio y utilizar la web en un entorno real.

Para ello, utilizando el apartado “**Migrar Web**” en el panel de control de CDMON, hemos debido introducir el usuario con dicho contrato (en este caso, nuestro cliente) y de esta

manera, se migra todo el contenido de nuestro hosting de pruebas gratuito al hosting pago de nuestro cliente, llegando así al final de la fase de despliegue.

Una vez desplegada la aplicación y en funcionamiento el ciclo de vida de nuestra aplicación continúa, aunque la parte del desarrollo esté finalizada y las siguientes fases estén fuera de nuestro alcance dentro del ciclo, la siguiente fase es la de **formación**, en esta fase ofrecemos una capacitación inicial a los usuarios y comprobamos su experiencia y resolvemos las posibles dificultades que puedan surgir, en esta fase lo único que podemos hacer dentro de nuestra limitación temporal es ofrecer una pestaña de “Ayuda” al usuario, mostrando como es el uso de nuestra aplicación y resolviendo las posibles dudas que prevemos que puedan surgir.

Por último, nos queda el **mantenimiento**, esta fase es una de las claves de éxito de nuestra aplicación, en esta fase minimizamos los pequeños errores que puedan ir surgiendo ya que los mismos surgen en toda aplicación viva.

Confirmamos el buen funcionamiento, la eficiencia y usabilidad del proyecto, ya que este está completado y lo mantenemos para garantizar que continúe funcionando correctamente en el futuro.

8. Conclusiones, mejoras futuras

Durante el desarrollo de este proyecto hemos buscado desarrollar una aplicación que facilite al centro la reserva de los espacios comunes.

Para lograr este fin en el menor tiempo posible nos hemos dividido las tareas a realizar y hemos ido realizando reuniones para ir valorando el avance del proyecto.

Al final del desarrollo de la aplicación tenemos como producto final un sistema que no solo permite la gestión de reservas de aulas por parte de todos los profesores con un par de clicks sino que también ofrece las estadísticas de las zonas más utilizadas de forma que el centro tome estos datos para mejoras del centro.

Esta aplicación, pese a cumplir completamente los requisitos establecidos en este proyecto, aún puede ser **mejorada** en ciertos aspectos:

- Implementación de una sub-funcionalidad (como lo puede ser, la gestión de los días libres de los profesores) obteniendo así una aplicación con la información centralizada y una mejor gestión de la misma.
- Desarrollar una aplicación para móviles, con opciones de descarga tanto para iOS como para Android, ofreciendo así a los usuarios la mejor experiencia si acceden desde un dispositivo móvil.
- Mejora a nivel código desacoplando la parte front de la parte back para que el desarrollo de la aplicación móvil fuese más integral, creando una API.

9. Bibliografía y referencias

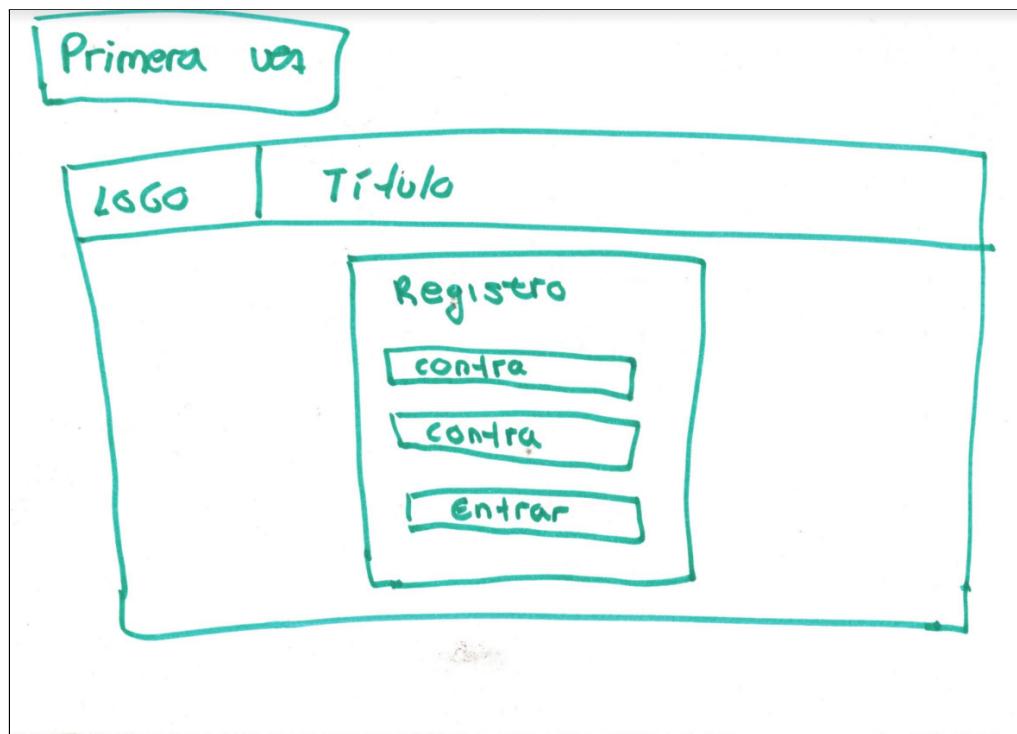
- *Bootstrap Icons*. (2022). Bootstrap Icons. <https://icons.getbootstrap.com/>
- Otto, M. J. T. (2022). *Introduction*. Bootstrap.
<https://getbootstrap.com/docs/5.0/getting-started/introduction/>
- *W3Schools Free Online Web Tutorials*. (2022). W3Schools.
<https://www.w3schools.com/>
- *SweetAlert*. (2022). Sweet Alert. <https://sweetalert.js.org/guides/>
- *Manual de Php*. (2022). <https://www.php.net/manual/es/index.php>
- P. (2022). *GitHub - PHPMailer/PHPMailer: The classic email sending library for PHP*. GitHub. <https://github.com/PHPMailer/PHPMailer>
- *El Tutorial de JavaScript Moderno*. (2022). El Tutorial de JavaScript Moderno.
<https://es.javascript.info/>

10. Anexos

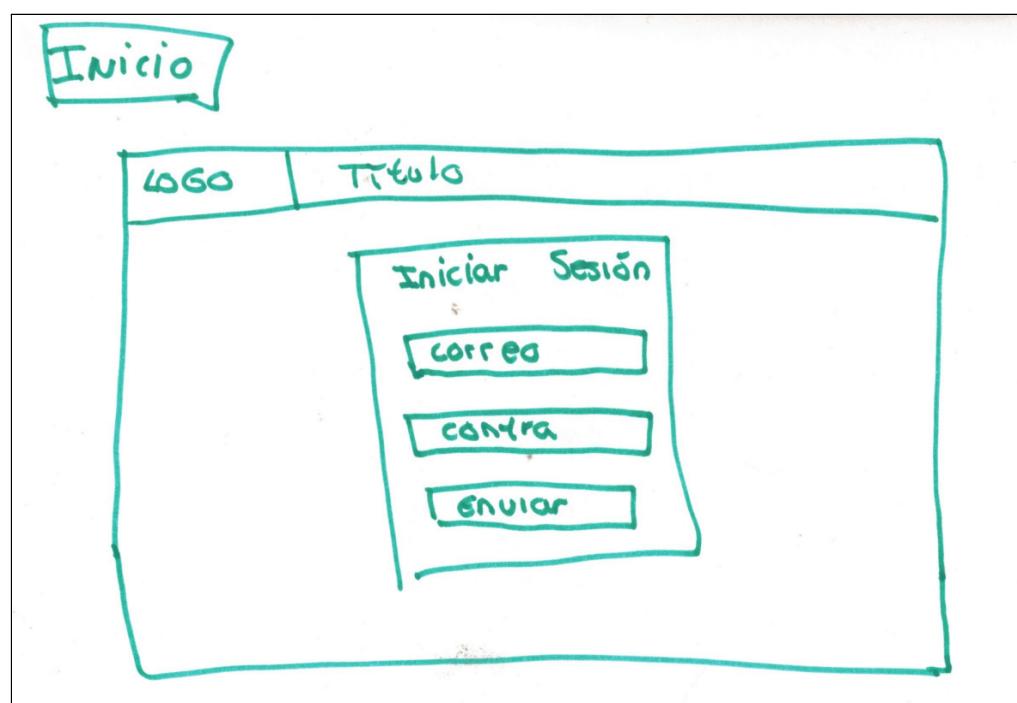
Glosario de términos

- **SGEC:** Sistema de Gestión de Espacios Comunes
- **PMV:** Producto mínimo viable
- **Crud:** Tablas que muestran los datos de la base de datos y que permite modificar, consultar, crear y borrar datos de estas (Create, read, update and delete, en inglés)
- **BBDD:** Base de datos
- **Cookie:** Pequeños fragmentos de texto que los sitios web que visitas envían al navegador. Permiten que los sitios web recuerden información sobre tu visita, lo que puede hacer que sea más fácil volver a visitar los sitios y hacer que estos te resulten más útiles.
- **PaaS:** Modalidad del cloud computing en la cual un tercero brinda el sistema de hardware y una plataforma de software de aplicaciones (Platform As a Service)

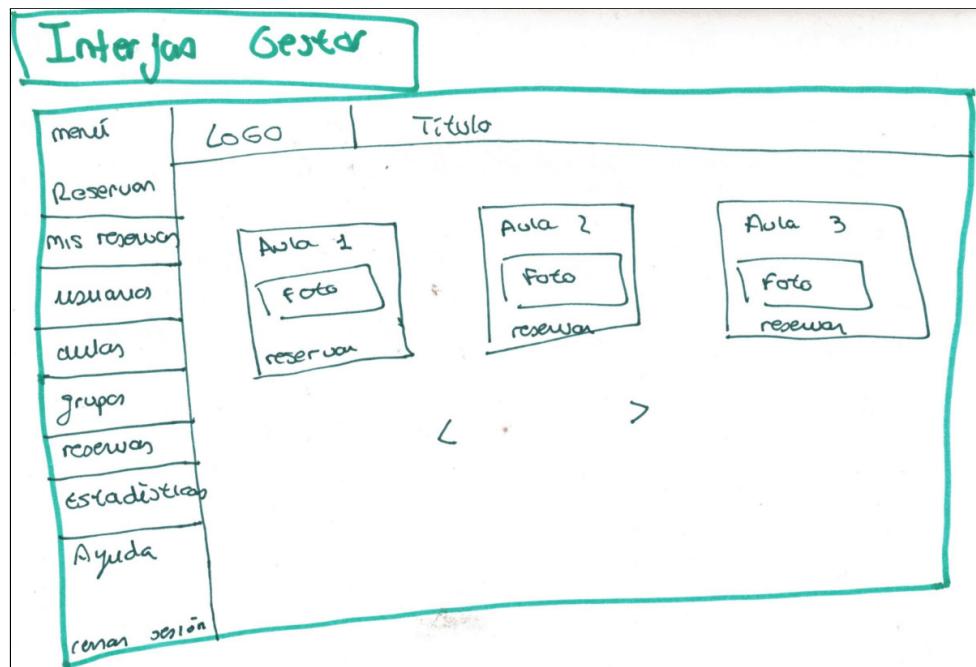
Anexo 1 (Boceto):



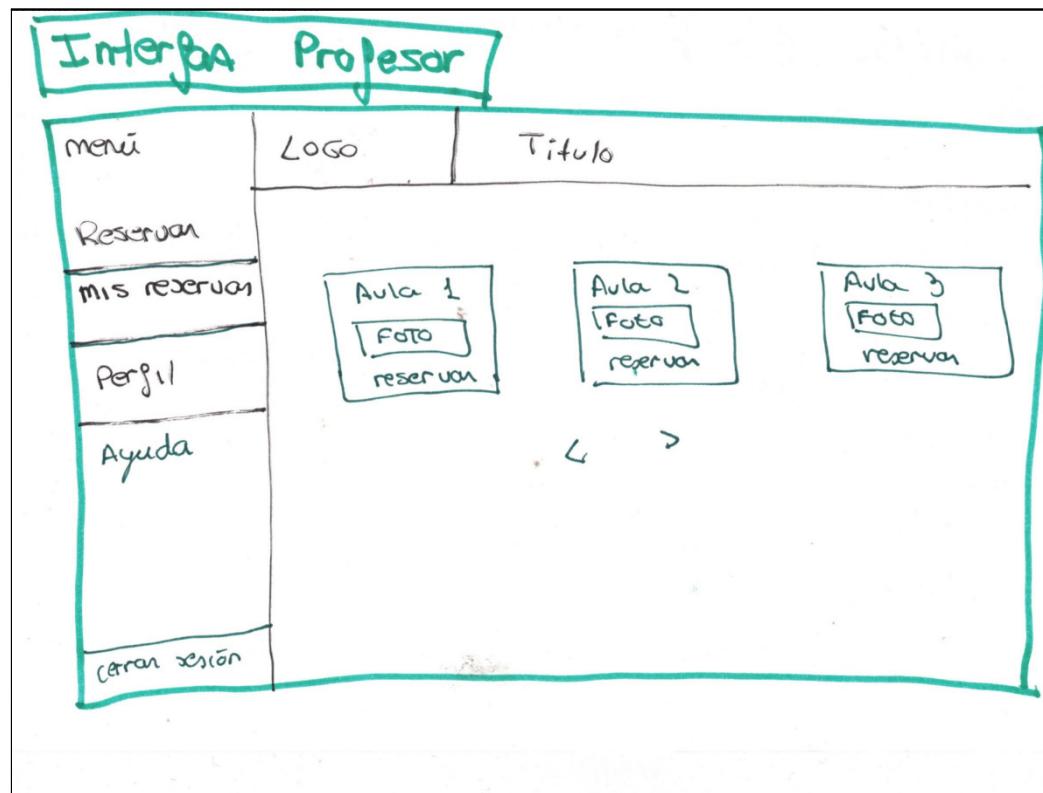
Boceto inicio de sesión



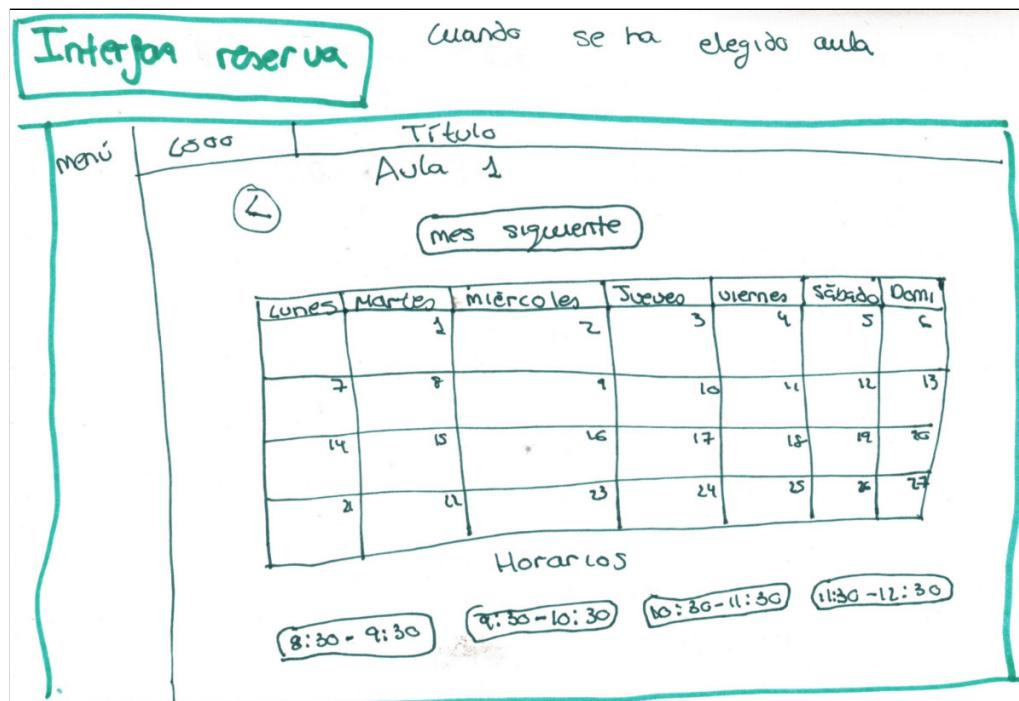
Boceto de primer inicio de sesión



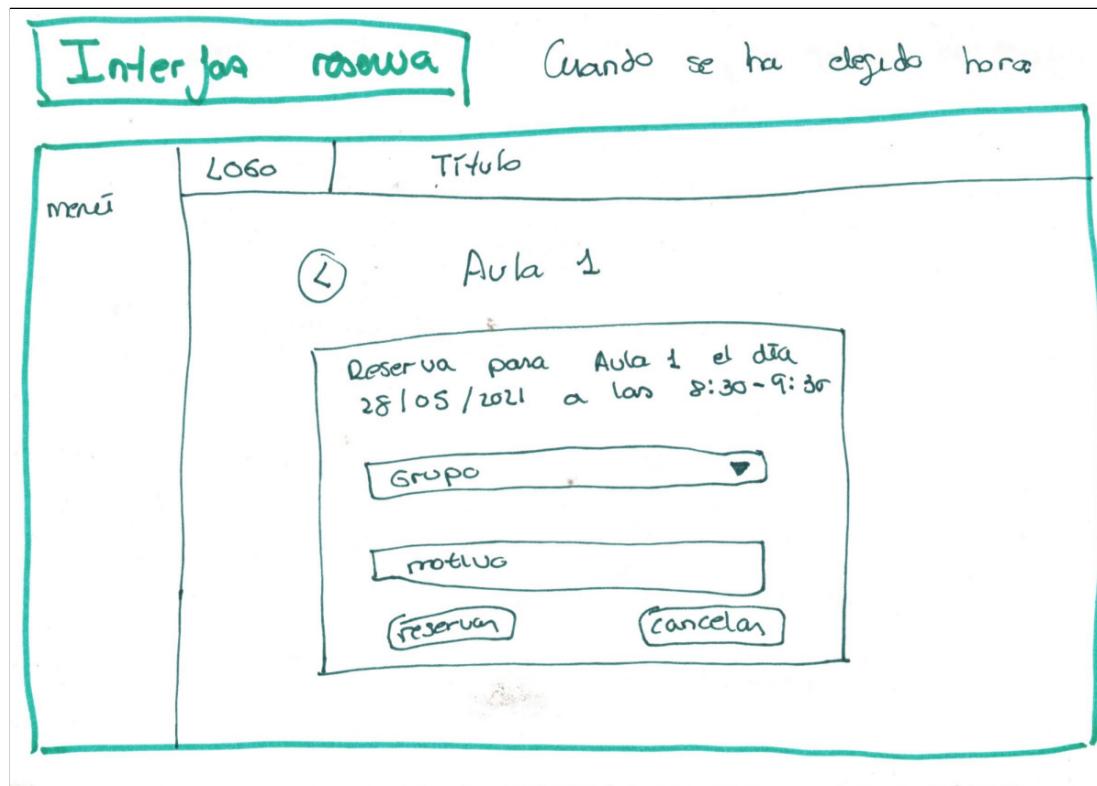
Boceto interfaz principal para rol “Gestor”



Boceto interfaz principal para rol “Profesor”



Boceto interfaz reserva de vista mensual una vez se ha elegido aula y día



Boceto interfaz reserva una vez se ha elegido hora.

Anexos 2 (Maqueta digital):



Maqueta digital para el inicio de sesión



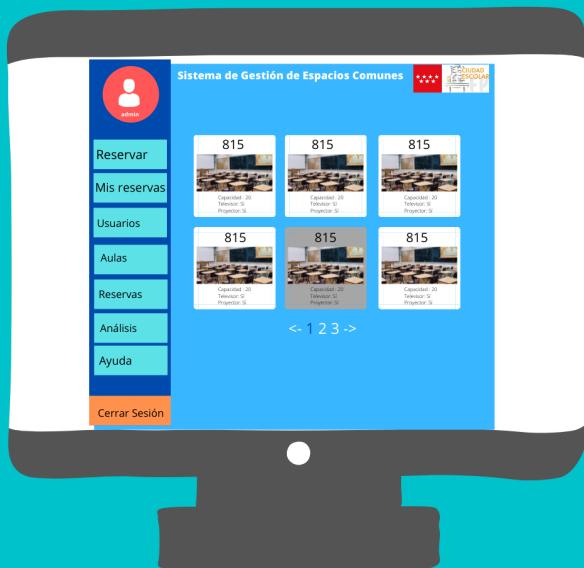
Maqueta digital para la primera vez que se inicia sesión

INTERFAZ PEC (Común Admin, Gestor)



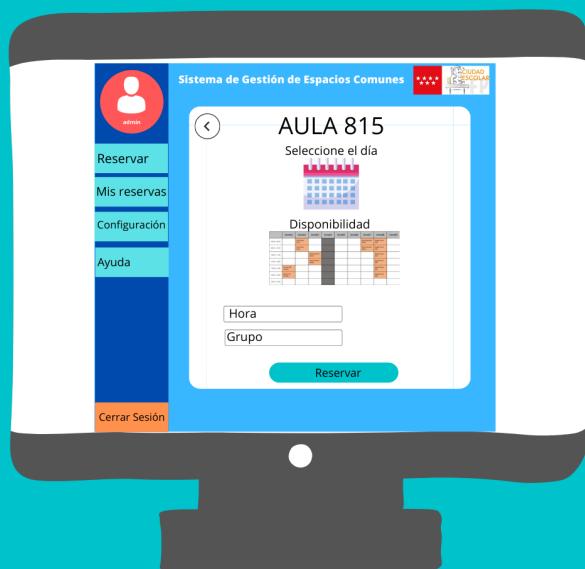
Maqueta digital para la interfaz de reservas visto desde el rol “Profesor”

INTERFAZ Gestor, Admin

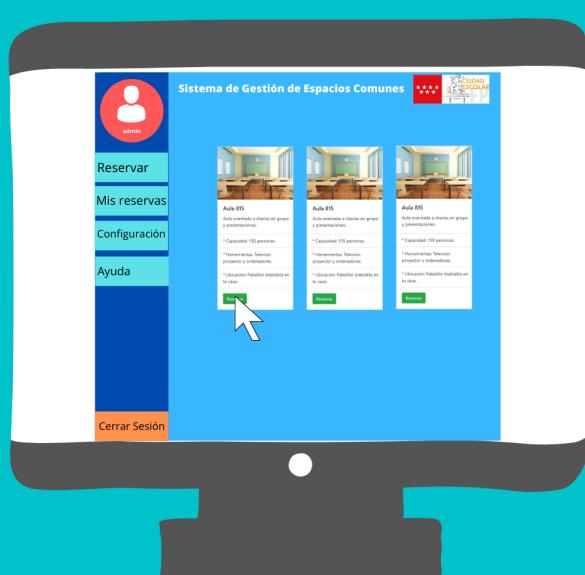


Maqueta digital para la interfaz de reservas visto desde el rol “Gestor”

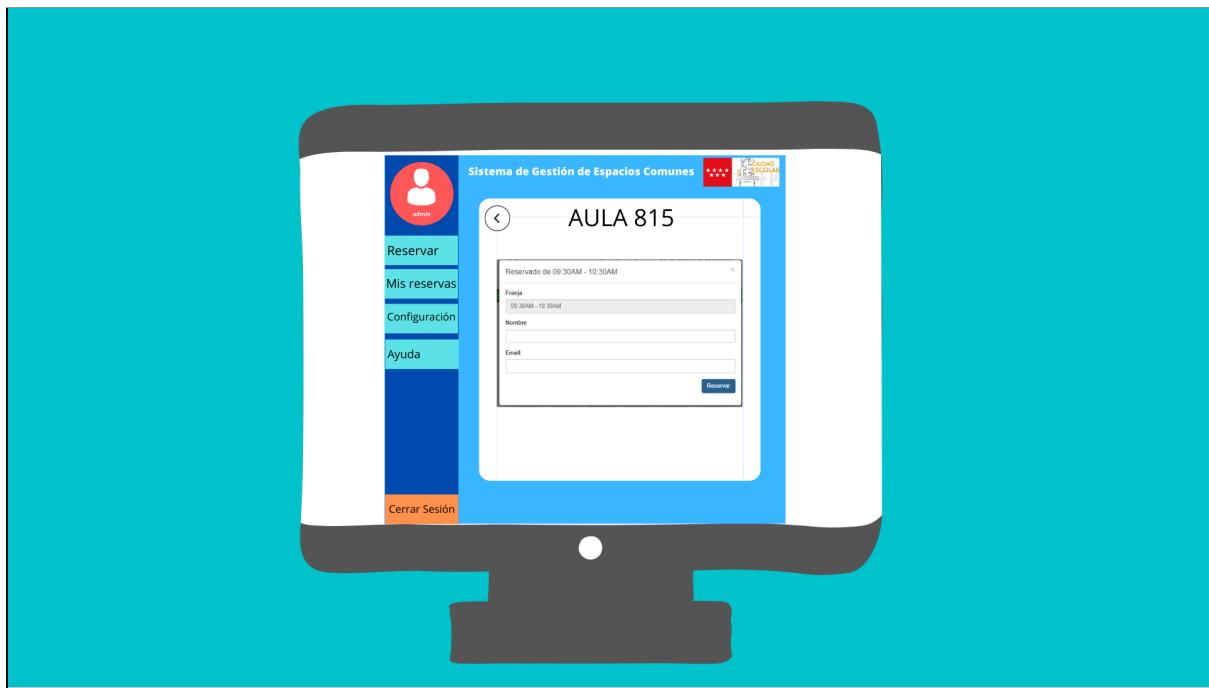
INTERFAZ RESERVA



Maqueta digital para la interfaz de reservas desde la vista semanal una vez se ha elegido aula



Maqueta digital para la interfaz principal donde se muestran las aulas

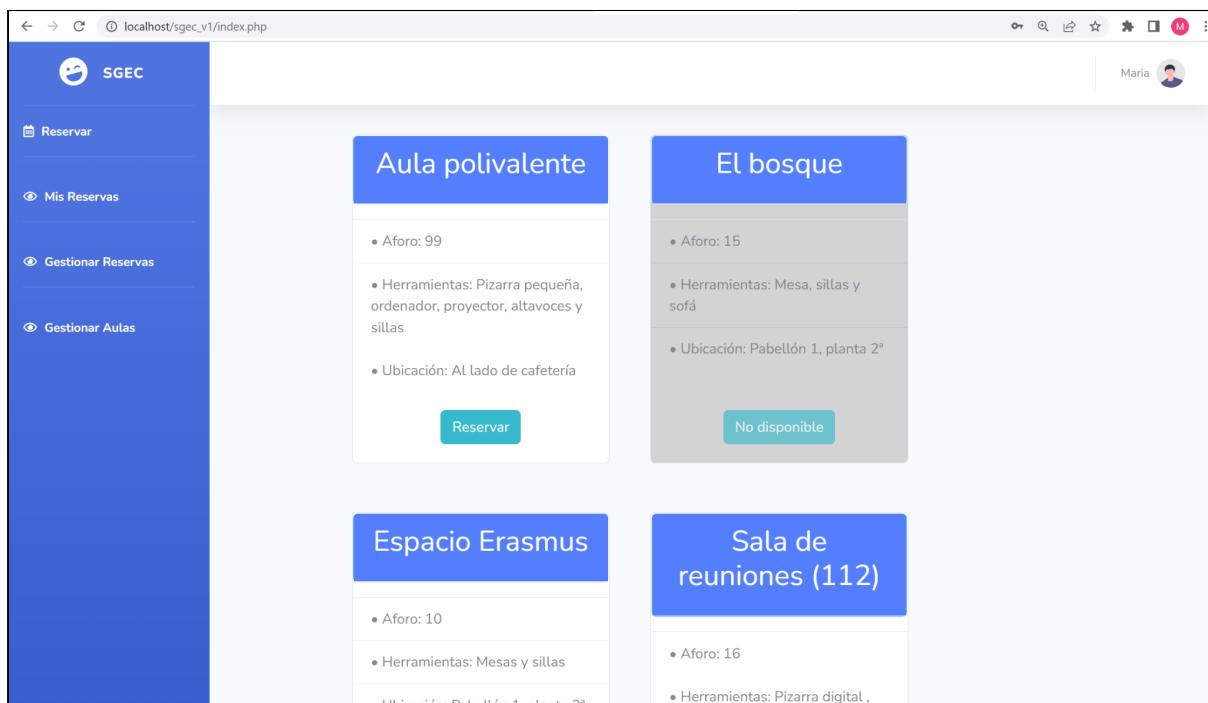


Maqueta digital para la interfaz de reservas desde la vista mensual una vez se ha elegido aula, día y hora.

Anexo 3 (Versión 1 de la aplicación):



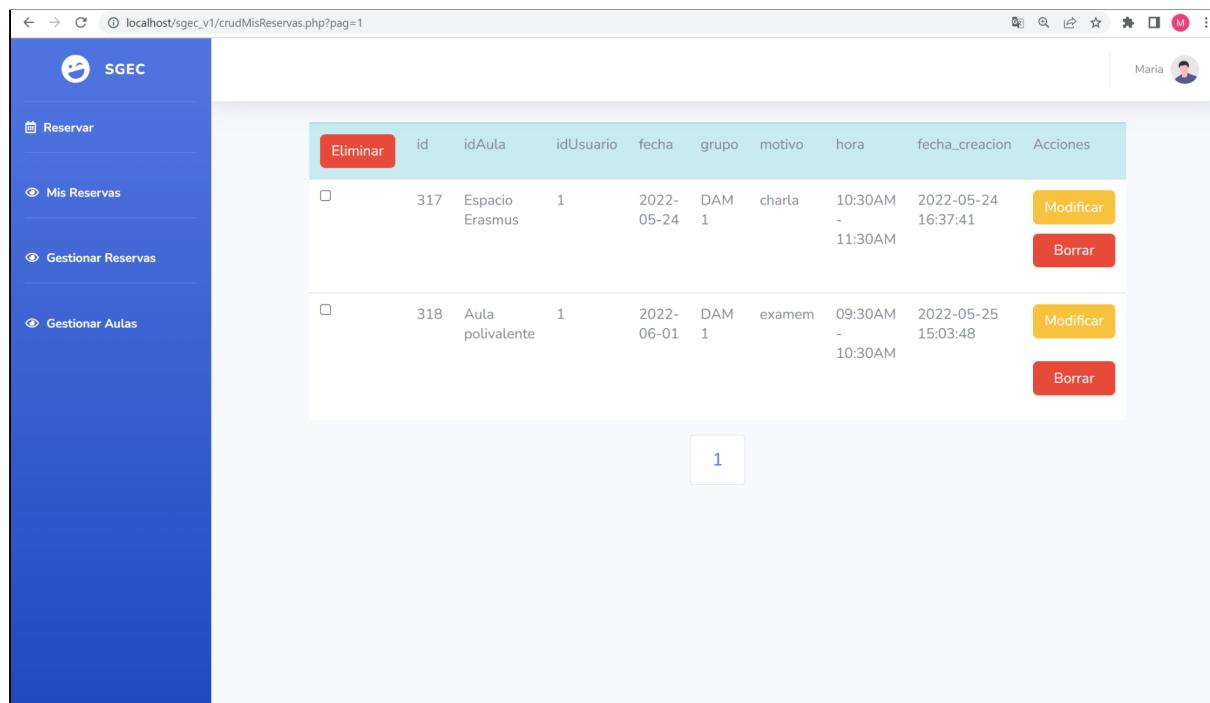
Primera versión del inicio de sesión



The screenshot shows a web browser window with the URL "localhost:sgec_v1/index.php". The left sidebar has a blue header "SGEC" and navigation items: "Reservar", "Mis Reservas", "Gestionar Reservas", and "Gestionar Aulas". The main content area displays four reservation cards:

- Aula polivalente**
 - Aforo: 99
 - Herramientas: Pizarra pequeña, ordenador, proyector, altavoces sillas
 - Ubicación: Al lado de cafeteríaReservar
- El bosque**
 - Aforo: 15
 - Herramientas: Mesa, sillas y sofá
 - Ubicación: Pabellón 1, planta 2^aNo disponible
- Espacio Erasmus**
 - Aforo: 10
 - Herramientas: Mesas y sillas
 - Ubicación: Pabellón 1, planta 2^a
- Sala de reuniones (112)**
 - Aforo: 16
 - Herramientas: Pizarra digital ,

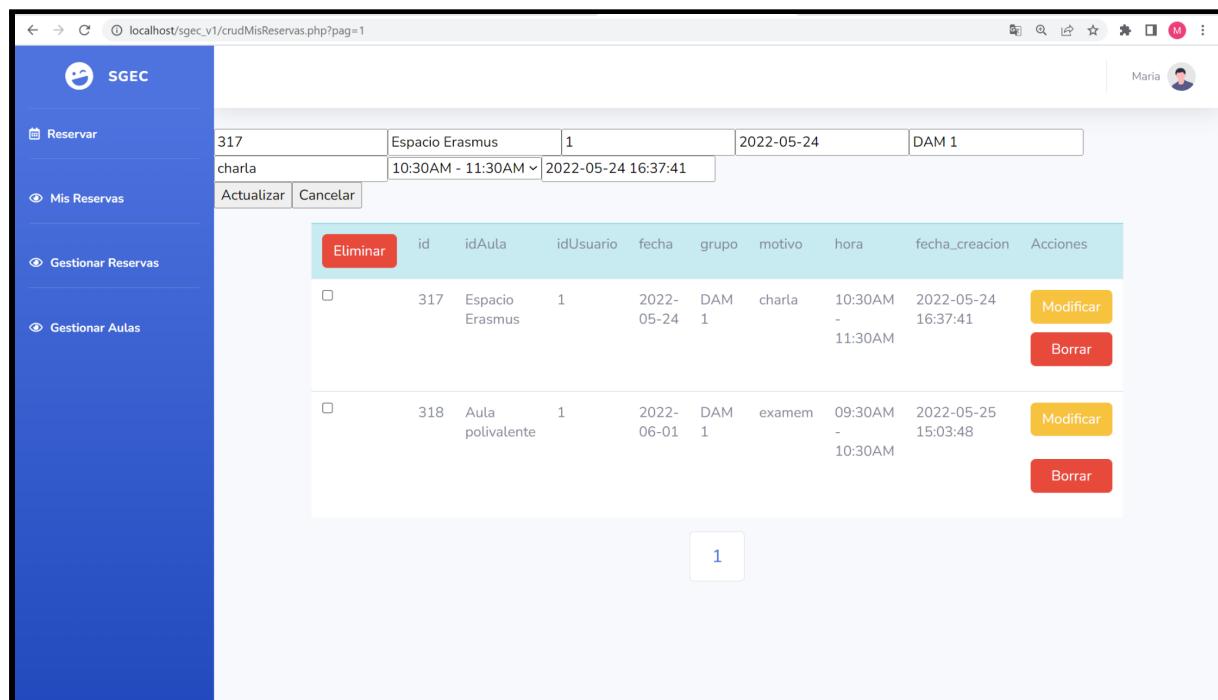
Primera versión de la interfaz de reservas



Eliminar	id	idAula	idUsuario	fecha	grupo	motivo	hora	fecha_creacion	Acciones
<input type="checkbox"/>	317	Espacio Erasmus	1	2022-05-24	DAM 1	charla	10:30AM - 11:30AM	2022-05-24 16:37:41	<button>Modificar</button> <button>Borrar</button>
<input type="checkbox"/>	318	Aula polivalente	1	2022-06-01	DAM 1	examem	09:30AM - 10:30AM	2022-05-25 15:03:48	<button>Modificar</button> <button>Borrar</button>

1

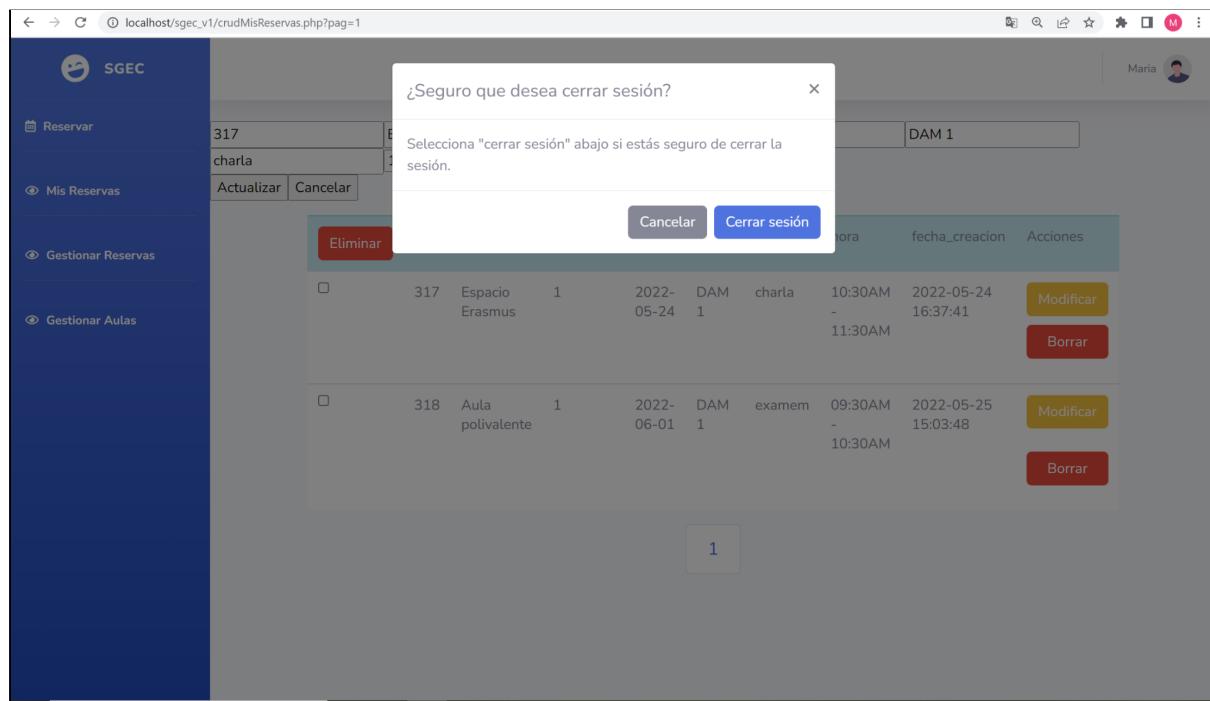
Primera versión de la interfaz de gestionar “Mis reservas”



Eliminar	id	idAula	idUsuario	fecha	grupo	motivo	hora	fecha_creacion	Acciones
<input type="checkbox"/>	317	Espacio Erasmus	1	2022-05-24	DAM 1	charla	10:30AM - 11:30AM	2022-05-24 16:37:41	<button>Modificar</button> <button>Borrar</button>
<input type="checkbox"/>	318	Aula polivalente	1	2022-06-01	DAM 1	examem	09:30AM - 10:30AM	2022-05-25 15:03:48	<button>Modificar</button> <button>Borrar</button>

1

Primera versión de la interfaz de modificar reserva

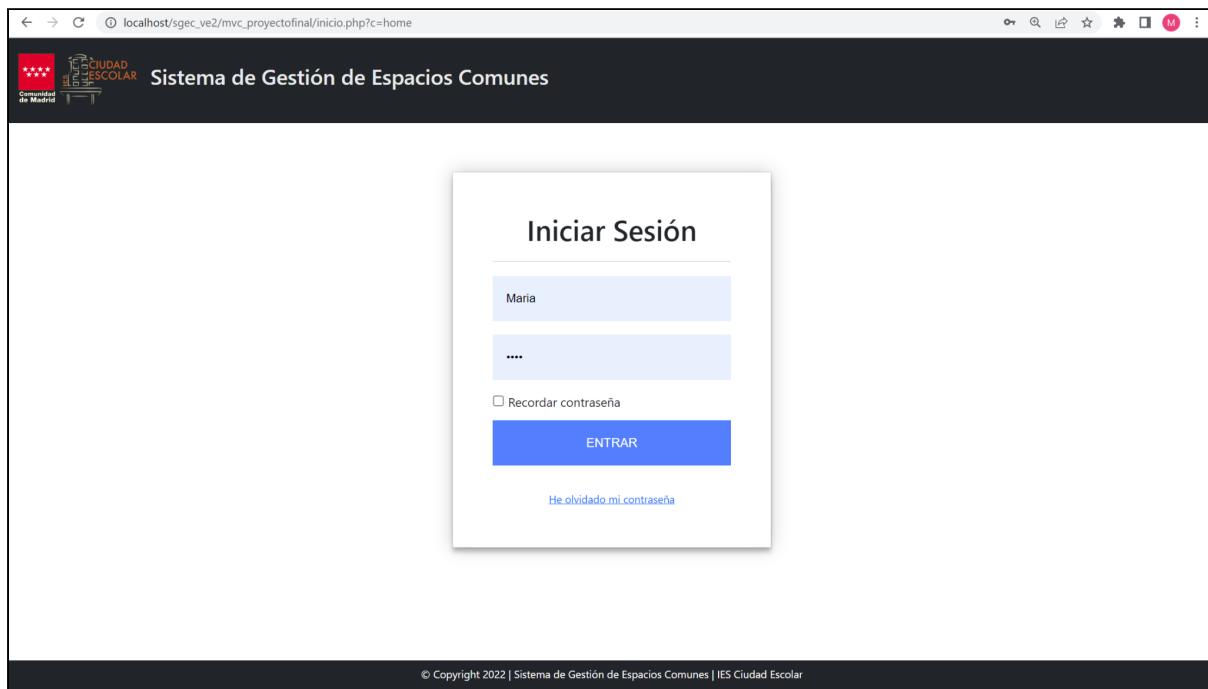


The screenshot shows a web-based application interface for managing common spaces. On the left, a sidebar menu includes 'Reservar', 'Mis Reservas', 'Gestionar Reservas', and 'Gestionar Aulas'. The main content area displays a list of reservations. A modal dialog box is centered over the page, asking '¿Seguro que desea cerrar sesión?' (Are you sure you want to close the session?). It contains instructions to select 'Cerrar sesión' if the user is sure. Below the dialog is a table listing two reservation entries:

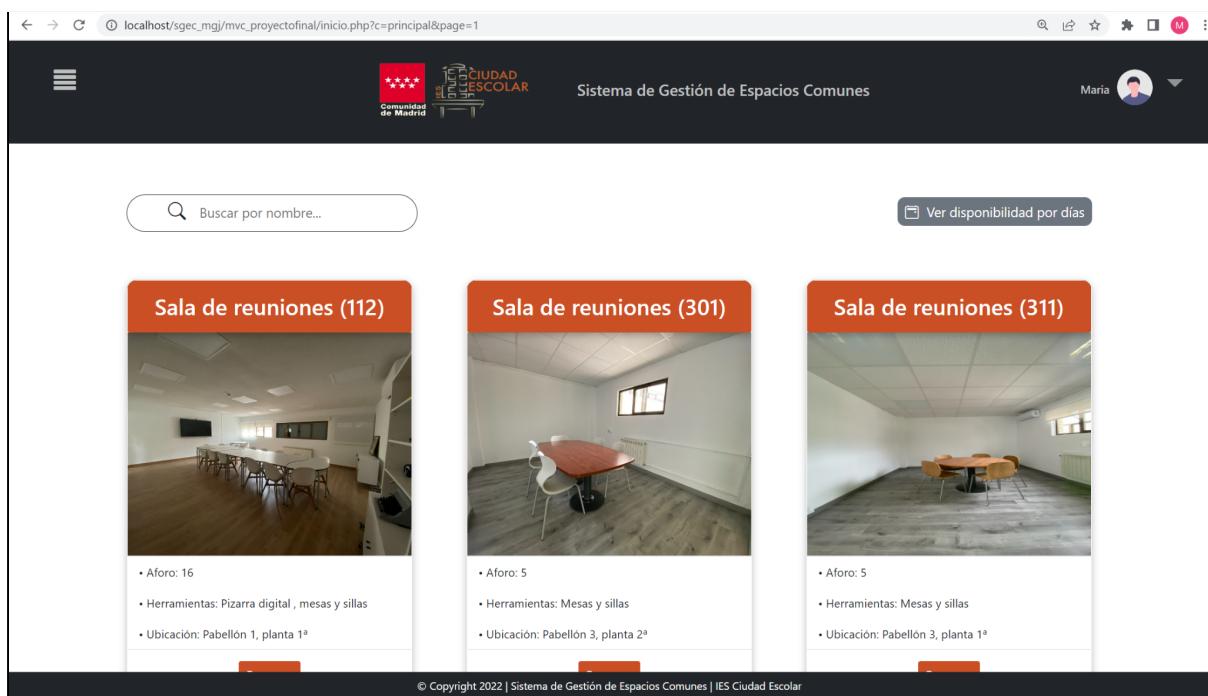
	ID	Espacio	Capacidad	Fecha	Hora	Descripción	Horario	fecha_creacion	Acciones
<input type="checkbox"/>	317	Espacio Erasmus	1	2022-05-24	DAM 1	charla	10:30AM - 11:30AM	2022-05-24 16:37:41	<button>Modificar</button> <button>Borrar</button>
<input type="checkbox"/>	318	Aula polivalente	1	2022-06-01	DAM 1	examem	09:30AM - 10:30AM	2022-05-25 15:03:48	<button>Modificar</button> <button>Borrar</button>

Primera versión de la interfaz de cierre de sesión

Anexo 4 (Versión 2 de la aplicación):



Segunda versión del inicio de sesión



The screenshot displays a list of three classrooms available for booking:

- Sala de reuniones (112)**: A room with a large conference table and chairs. Details: Aforo: 16, Herramientas: Pizarra digital, mesas y sillas, Ubicación: Pabellón 1, planta 1^a.
- Sala de reuniones (301)**: A room with a smaller conference table and chairs. Details: Aforo: 5, Herramientas: Mesas y sillas, Ubicación: Pabellón 3, planta 2^a.
- Sala de reuniones (311)**: An empty room with a small round table and chairs. Details: Aforo: 5, Herramientas: Mesas y sillas, Ubicación: Pabellón 3, planta 1^a.

Segunda versión de la interfaz donde se muestran las aulas

localhost/sgec_mgj/mvc_proyectofinal/inicio.php?c=calendario&id=Aula%20polivalente

 Sistema de Gestión de Espacios Comunes

Maria

[Ver aulas](#)

Reserva para el Aula polivalente

< Mayo 2022 >

[Vista semanal](#)

Lunes	Martes	Miércoles	Jueves	Viernes	Sábado	Domingo
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					
Horarios	Horarios					

© Copyright 2022 | Sistema de Gestión de Espacios Comunes | IES Ciudad Escolar

Segunda versión de la vista mensual

localhost/sgec_mgj/mvc_proyectofinal/inicio.php?c=calendarioSemanal&id=Aula%20polivalente&week=22&year=2022

[Ver aulas](#)

Reserva para el Aula polivalente

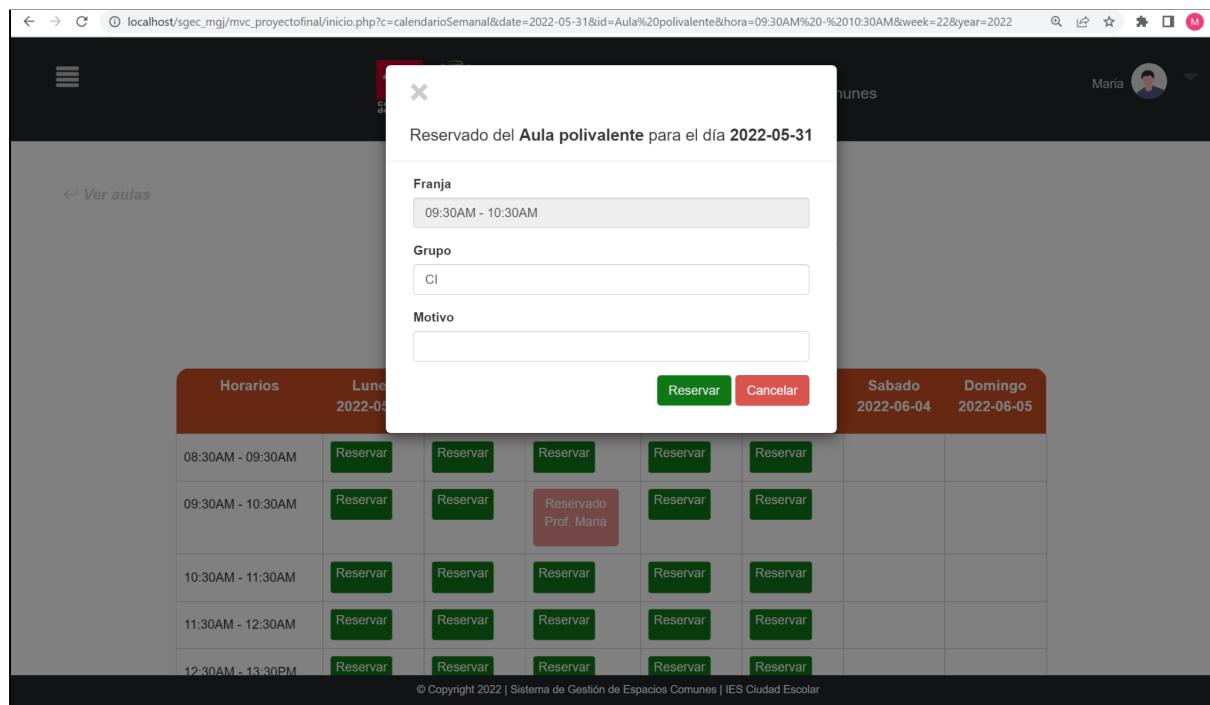
< Semana del 2022-05-30 >

[Vista mensual](#)

Horarios	Lunes 2022-05-30	Martes 2022-05-31	Miércoles 2022-06-01	Jueves 2022-06-02	Viernes 2022-06-03	Sábado 2022-06-04	Domingo 2022-06-05
08:30AM - 09:30AM	Reservar	Reservar	Reservar	Reservar	Reservar		
09:30AM - 10:30AM	Reservar	Reservar	Reservado Prof. María	Reservar	Reservar		
10:30AM - 11:30AM	Reservar	Reservar	Reservar	Reservar	Reservar		
11:30AM - 12:30AM	Reservar	Reservar	Reservar	Reservar	Reservar		
12:30AM - 13:30PM	Reservar	Reservar	Reservar	Reservar	Reservar		
13:30PM - 14:30PM	Reservar	Reservar	Reservar	Reservar	Reservar		

© Copyright 2022 | Sistema de Gestión de Espacios Comunes | IES Ciudad Escolar

Segunda versión de la vista semanal



Reservado del **Aula polivalente** para el día **2022-05-31**

Franja
09:30AM - 10:30AM

Grupo
CI

Motivo

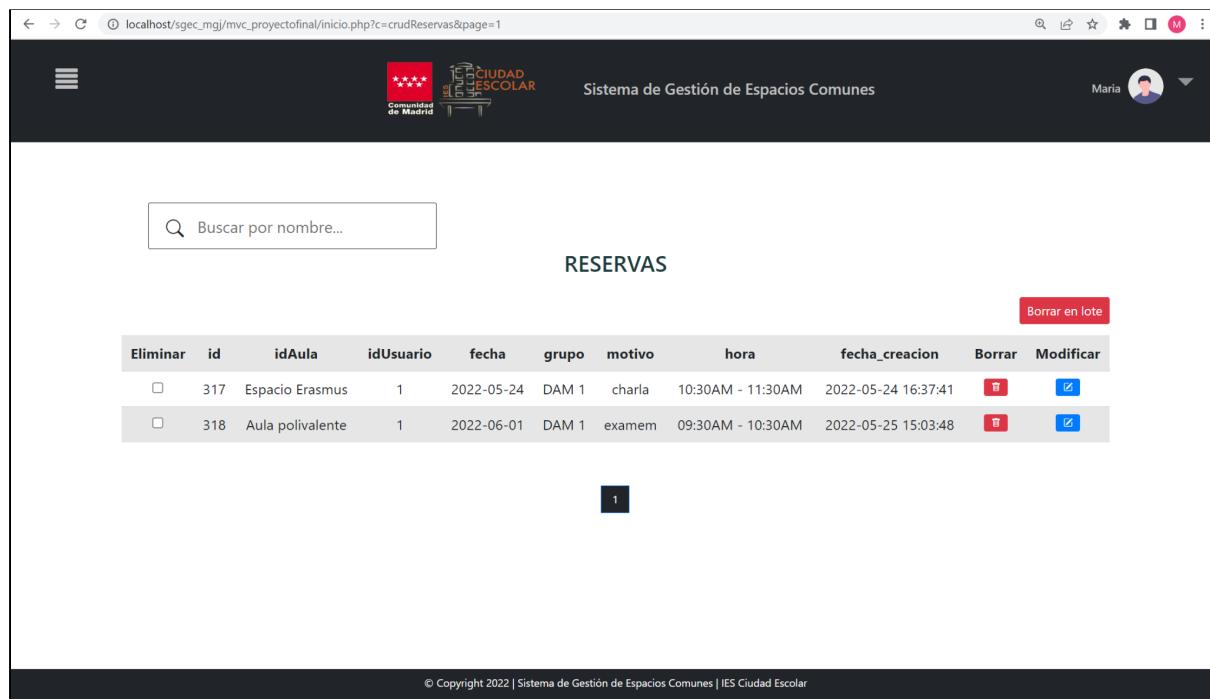
Horarios Lunes 2022-05-31

Horarios	08:30AM - 09:30AM	09:30AM - 10:30AM	10:30AM - 11:30AM	11:30AM - 12:30AM	12:30AM - 13:30PM
Reservar	Reservar	Reservar	Reservar	Reservar	Reservar
Reservar	Reservar	Reservado Prof. Marea	Reservar	Reservar	Reservar
Reservar	Reservar	Reservar	Reservar	Reservar	Reservar
Reservar	Reservar	Reservar	Reservar	Reservar	Reservar

Sabado 2022-06-04 Domingo 2022-06-05

© Copyright 2022 | Sistema de Gestión de Espacios Comunes | IES Ciudad Escolar

Segunda versión de la interfaz para la vista mensual y semanal una vez se ha elegido aula, día y hora



RESERVAS

Eliminar	id	idAula	idUsuario	fecha	grupo	motivo	hora	fecha_creacion	Borrar	Modificar
<input type="checkbox"/>	317	Espacio Erasmus	1	2022-05-24	DAM 1	charla	10:30AM - 11:30AM	2022-05-24 16:37:41		
<input type="checkbox"/>	318	Aula polivalente	1	2022-06-01	DAM 1	examem	09:30AM - 10:30AM	2022-05-25 15:03:48		

1

© Copyright 2022 | Sistema de Gestión de Espacios Comunes | IES Ciudad Escolar

Segunda versión de la interfaz de gestión de reservas



localhost/sgec_mgj/mvc_proyectofinal/inicio.php?c=crudReservas&page=1

Maria

Buscar por nombre...

Eliminar	id	idAula	idUser
<input type="checkbox"/>	317	Espacio Erasmus	
<input type="checkbox"/>	318	Aula polivalente	

Borrar en lote

fecha_creacion	Borrar	Modificar
2022-05-24 16:37:41		
2022-05-25 15:03:48		

© Copyright 2022 | Sistema de Gestión de Espacios Comunes | IES Ciudad Escolar

Segunda versión de la interfaz de modificar reserva



localhost/sgec_mgj/mvc_proyectofinal/inicio.php?c=crudReservas&page=1

Maria

Reservar

Mis reservas

Gestionar

Reservas

Aulas

Usuarios

Roles

Grupos

Festivos

Estadísticas

Configuracion

Buscar por nombre...

RESERVAS

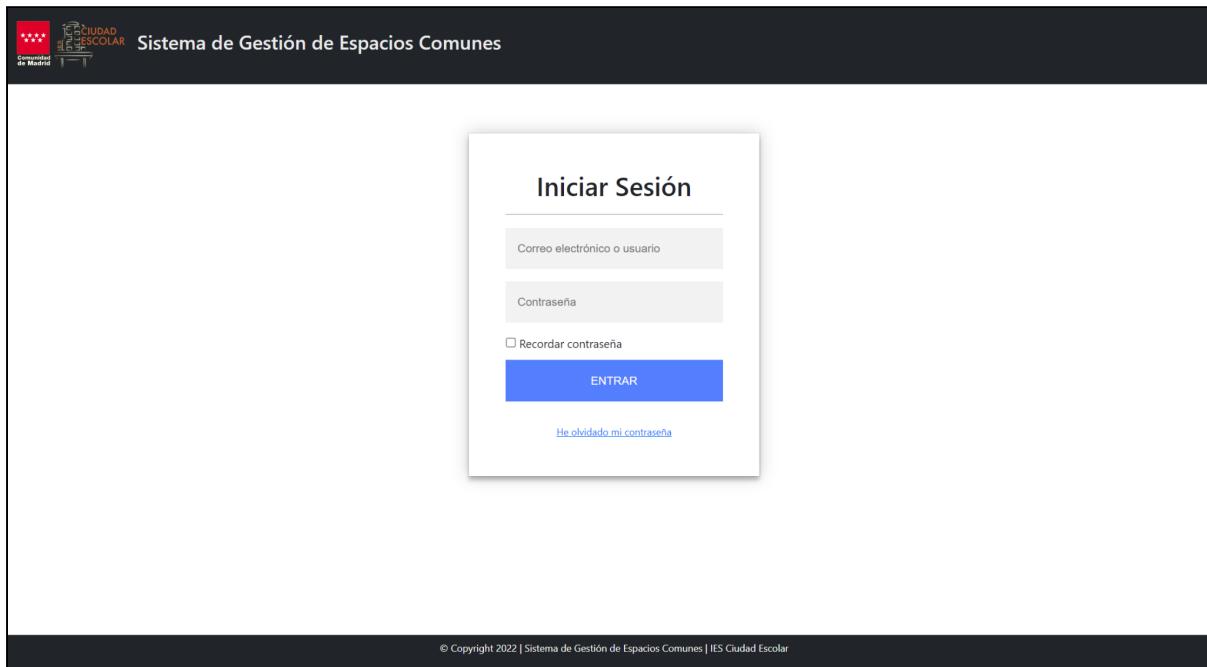
idAula	idUser	fecha	grupo	motivo	hora	fecha_creacion	Borrar	Modificar
Espacio Erasmus	1	2022-05-24	DAM 1	charla	10:30AM - 11:30AM	2022-05-24 16:37:41		
Aula polivalente	1	2022-06-01	DAM 1	examen	09:30AM - 10:30AM	2022-05-25 15:03:48		

Borrar en lote

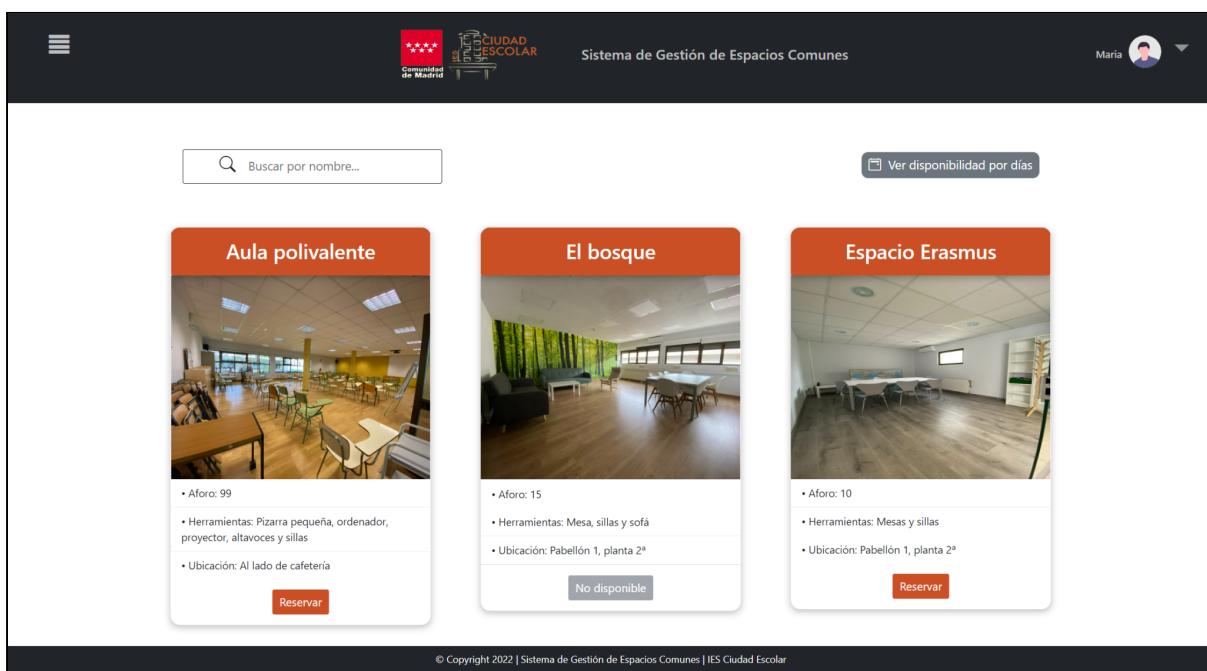
© Copyright 2022 | Sistema de Gestión de Espacios Comunes | IES Ciudad Escolar

Segunda versión del menú

Anexo 5 (Versión final de la aplicación):



Versión final del inicio de sesión



The screenshot shows the main interface of the SGEC system. At the top, it features the 'Comunidad de Madrid' logo, the text 'Sistema de Gestión de Espacios Comunes', and a user profile icon labeled 'Maria'. Below this is a search bar with the placeholder 'Buscar por nombre...' and a button 'Ver disponibilidad por días'. The main content area displays three classroom cards:

- Aula polivalente**: Shows a photo of a multi-purpose room with tables and chairs. Details: Aforo: 99, Herramientas: Pizarra pequeña, ordenador, proyector, altavoces y sillas, Ubicación: Al lado de cafetería. A 'Reservar' (Reserve) button is at the bottom.
- El bosque**: Shows a photo of a room decorated to look like a forest. Details: Aforo: 15, Herramientas: Mesa, sillas y sofá, Ubicación: Pabellón 1, planta 2^a. A 'No disponible' (Not available) button is at the bottom.
- Espacio Erasmus**: Shows a photo of a room with white tables and chairs. Details: Aforo: 10, Herramientas: Mesas y sillas, Ubicación: Pabellón 1, planta 2^a. A 'Reservar' (Reserve) button is at the bottom.

At the bottom of the page is a footer with the copyright notice '© Copyright 2022 | Sistema de Gestión de Espacios Comunes | IES Ciudad Escolar'.

Versión final de la interfaz donde se muestran las aulas

Comunidad de Madrid

[← Ver aulas](#)

Reserva para el Aula polivalente

< Mayo 2022 >

[Vista semanal](#) [Vista diaria](#)

Lunes	Martes	Miércoles	Jueves	Viernes	Sábado	Domingo
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					
∅ Ver disponibilidad		∅ Ver disponibilidad				

© Copyright 2022 | Sistema de Gestión de Espacios Comunes | IES Ciudad Escolar

Versión final de la vista mensual

No es seguro | sgec.es.malias.net/?c=calendario&date=2022-05-31&id=Sala%20de%20reuniones%20(112)&month=05&year=2022

Lunes	Martes	Miércoles	Jueves	Viernes	Sábado	Domingo
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					
∅ Ver disponibilidad		∅ Ver disponibilidad				

Horarios del día: 31/05/2022

08:30AM - 09:30AM 09:30AM - 10:30AM 10:30AM - 11:30AM 11:30AM - 12:30PM 12:30AM - 13:30PM 13:30PM - 14:30PM

● Disponible
● Reservado

© Copyright 2022 | Sistema de Gestión de Espacios Comunes | IES Ciudad Escolar

Versión final de la vista mensual una vez se ha elegido aula

Lunes Martes Miércoles Jueves Viernes Sábado Domingo

2	3				1	
9	10				7	8
16	17				14	15
23	24				21	22
30	31				28	29
Ver disponibilidad				Reservar Cancelar		

Reservado de 10:30AM - 11:30AM

Franja
10:30AM - 11:30AM

Grupo
AC 1

Motivo

Horarios del día: 31/05/2022

08:30AM - 09:30AM 09:30AM - 10:30AM 10:30AM - 11:30AM 11:30AM - 12:30AM 12:30AM - 13:30PM 13:30PM - 14:30PM

● Disponible
● Reservado

© Copyright 2022 | Sistema de Gestión de Espacios Comunes | IES Ciudad Escolar

Versión final de la vista mensual una vez se ha elegido aula, día y hora

Ver aulas

Reserva para el Sala de reuniones (112)

< Semana del 30-05-2022 >

[Vista mensual](#) [Vista diaria](#)

Horarios	Lunes 30-05-2022	Martes 31-05-2022	Miércoles 01-06-2022	Jueves 02-06-2022	Viernes 03-06-2022	Sábado 04-06-2022	Domingo 05-06-2022
08:30AM - 09:30AM	Reservar	Reservar	Reservar	No disponible Mejora	No disponible Mejora		
09:30AM - 10:30AM	Reservar	Reservar	Reservar	No disponible Mejora	No disponible Mejora		
10:30AM - 11:30AM	Reservar	Reservar	Reservar	No disponible Mejora	No disponible Mejora		
11:30AM - 12:30AM	Reservar	Reservar	Reservar	No disponible Mejora	No disponible Mejora		
12:30AM - 13:30PM	Reservar	Reservar	Reservar	No disponible Mejora	No disponible Mejora		

© Copyright 2022 | Sistema de Gestión de Espacios Comunes | IES Ciudad Escolar

Versión final de la vista semanal

[Ver aulas](#)

< Horarios del día Lunes 30-05-2022 >

Horarios	Aula polivalente	Espacio Erasmus	Sala de reuniones (112)	Sala de reuniones (301)	Sala de reuniones (311)
08:30AM - 09:30AM	No disponible Reforma	Reservar	Reservar	Reservar	Reservar
09:30AM - 10:30AM	No disponible Reforma	Reservar	Reservar	Reservar	Reservar
10:30AM - 11:30AM	No disponible Reforma	Reservar	Reservar	Reservar	Reservar
11:30AM - 12:30AM	No disponible Reforma	Reservar	Reservar	Reservar	Reservar
12:30AM - 13:30PM	No disponible Reforma	Reservar	Reservar	Reservar	Reservar
13:30PM - 14:30PM	No disponible Reforma	Reservar	Reservar	Reservar	Reservar

© Copyright 2022 | Sistema de Gestión de Espacios Comunes | IES Ciudad Escolar

www.sgec.es.mialias.net/?c=principal

Versión final de la vista diaria

[Buscar por nombre...](#)

Usuarios

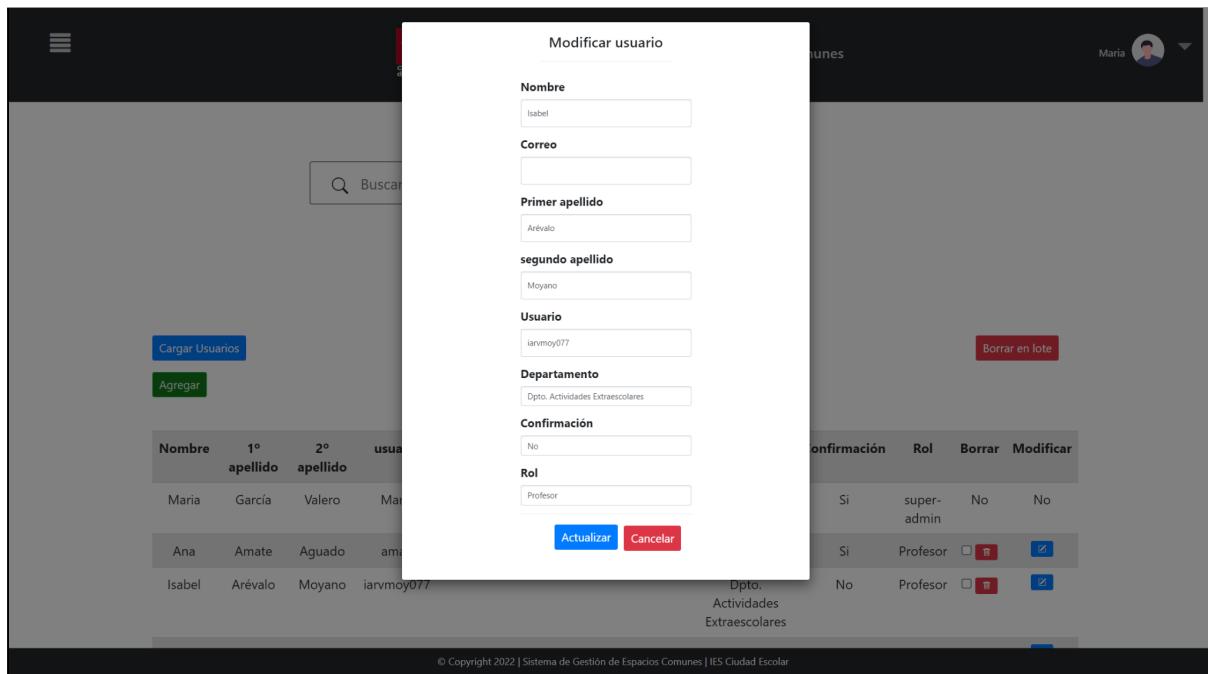
[Cargar Usuarios](#) [Borrar en lote](#)

[Agregar](#)

Nombre	1º apellido	2º apellido	usuario	correo	departamento	Confirmación	Rol	Borrar	Modificar
Maria	García	Valero	Maria	mariagarcia.daw@ciudadescolarf.es	Dpto. Informatica	Si	super-admin	No	No
Ana	Amate	Aguado	ama2	gerardopimentel.daw@ciudadescolarf.es	Dpto. Fol.	Si	Profesor	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Isabel	Arévalo	Moyano	iarvmo77		Dpto. Actividades Extraescolares	No	Profesor	<input type="checkbox"/>	<input checked="" type="checkbox"/>

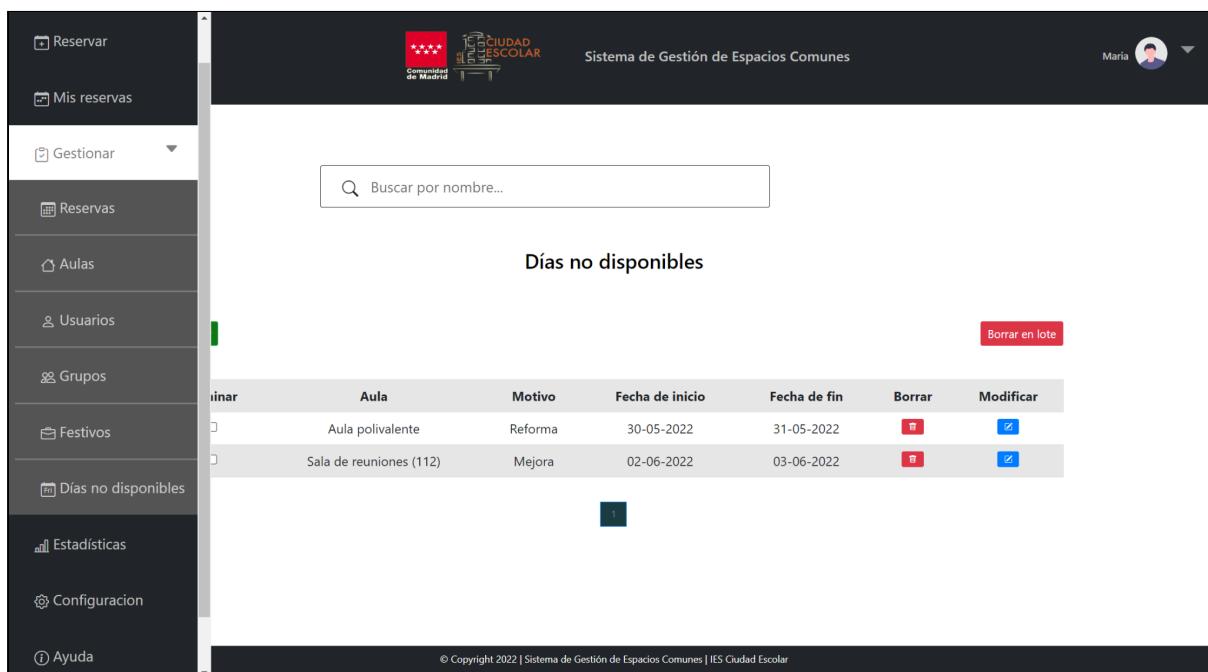
© Copyright 2022 | Sistema de Gestión de Espacios Comunes | IES Ciudad Escolar

Versión final del crud usuarios



Nombre	1º apellido	2º apellido	usuario
Maria	García	Valero	Mari
Ana	Amate	Aguado	amatea
Isabel	Arévalo	Moyano	iarvmoy077

Versión final de la interfaz de modificación de datos en el crud



Día	Aula	Motivo	Fecha de inicio	Fecha de fin	Borrar	Modificar
	Aula polivalente	Reforma	30-05-2022	31-05-2022	<input type="button" value="Borrar"/>	<input checked="" type="button" value="Modificar"/>
	Sala de reuniones (112)	Mejora	02-06-2022	03-06-2022	<input type="button" value="Borrar"/>	<input checked="" type="button" value="Modificar"/>

Versión final del menú para el rol “Gestor”

Anexo 6 (Cartel con Qr para reservar un aula):



Comunidad de Madrid IES CIUDAD ESCOLAR

Sistema de Gestión de Espacios Comunes

RESERVAR ESTA AULA

Para realizar la reserva de este aula escaneando el código QR, debe seguir los siguientes pasos:

- Asegurarse de haber iniciado sesión en nuestra web: "<https://www.educa2.madrid.org/web/centro.ies.ciudadescolar.madrid/sgec.com>"
- Una vez comprobado el paso anterior, debe abrir la cámara de su dispositivo móvil y dirigirla hacia el código QR. Esto le dará la opción de abrir el enlace, el cual debe pulsar.
- Ahora debe escoger el día y hora que desea entre los disponibles y listo, ¡ha realizado la reserva!



© Copyright 2022 | Sistema de Gestión de Espacios Comunes | IES Ciudad Escolar

Versión final del cartel con QR para reservar un aula

Anexo 7 (Diagrama de Gantt):

Fase 1/4

Fase 2/4

ACTIVIDAD	SEMANAS de 14/03/22 a 03/06/22										
	1	2	3	4	5	6	7	8	9	10	11
FASE 3											
Análisis y diseño											
Análisis de nuevos requisitos											
División de nuevos requisitos en tareas											
Diseño de nuevas vistas de la aplicación											
Desarrollo e implementación											
Mejora de funcionalidades anteriores											
Desarrollo de nuevas vistas y funcionalidades											
Despliegue											
Despliegue de prueba											
Pruebas											
Pruebas de seguridad y funcionamiento de la fase 3											
Pruebas de accesibilidad y usabilidad											
Documentación											
Desarrollo 100% documentación											
Entrega prototipo Fase 3											

Fase 3/4

ACTIVIDAD	SEMANAS de 14/03/22 a 03/06/22											
	1	2	3	4	5	6	7	8	9	10	11	12
FASE FINAL												
Desarrollo e implementación												
Mejora de funcionalidades anteriores												
Despliegue												
Migración y despliegue definitivo												
Pruebas												
Pruebas de seguridad y funcionamiento de la aplicación completa												
Pruebas de accesibilidad y usabilidad de la aplicación completa												
Entrega aplicación y documentación												

Fase 4/4

Anexo 8 (Manual de usuario):

En el pendrive que se ha entregado junto con este documento en el departamento de informática del IES Ciudad Escolar se adjunta un manual de usuario para profesores con todas las funcionalidades que pueden realizar.

Anexo 9 (Manual de gestor):

En el pendrive que se ha entregado junto con este documento en el departamento de informática del IES Ciudad Escolar se adjunta un manual de gestor para aquellos usuarios con rol “Gestor” donde se detallan todas las funcionalidades que pueden realizar.

Anexo 10 (Accesibilidad “Lighthouse”):

Resultado tras la comprobación de la accesibilidad con la extensión “Lighthouse”.

